Contents

Release notes and notices .......................................... 18
Version history .......................................................... 18
Release Notes ............................................................ 18
Flash ................................................................. 19
10.2 Enhancements ..................................................... 19
  No architecture changes required ............................... 20
  What's new in BMC Atrium Discovery version 10.2? .... 20
  CMDB Synchronization improvements ......................... 22
  Blade enclosure discovery ......................................... 22
  Management controller discovery ............................... 22
  Host/Storage linking ............................................... 23
  SAN Switch discovery .............................................. 24
  Miscellaneous ....................................................... 24
  Model changes ...................................................... 24
  Changes to Discovery Commands ................................. 31
Package list - BMC Atrium Discovery 10.2 ...................... 113
Out of the box discovery capabilities ......................... 117
Report changes ....................................................... 123
Changes to third party software license terms ............... 124
BMC Atrium Discovery copyright information .................. 126
Known and corrected issues ....................................... 178
  Defects resolved in this version ............................... 178
  Known defects in this version ................................. 211
Important information for users of BMC Atrium CMDB ........ 215
  Interaction of BMC Atrium Discovery with other BMC data providers 215
  Prerequisites for CMDB synchronization ...................... 216
  Co-existence with other data providers ..................... 217
Technical bulletins .................................................. 219
  Documentation changes since release ....................... 219
Product announcements ............................................. 219
  August 2014 Technology Knowledge Update and Operating System Upgrade packages 220
September 2014 Technology Knowledge Update and Operating System
Upgrade packages ............................................................. 221
October 2014 Technology Knowledge Update and Operating System
Upgrade packages ............................................................. 222
Accessing the release ....................................................... 223
Supported versions of BMC Atrium Discovery ....................... 224
Limitations and restrictions of this version ......................... 227
Operational warnings ....................................................... 227
Limitations and restrictions .............................................. 227
Getting started ............................................................... 239
Key concepts ................................................................. 239
Business Service Management ......................................... 239
Big Discovery ................................................................. 241
BMC Atrium Discovery .................................................... 242
Glossary ...................................................................... 246
End to end process ......................................................... 252
Understanding BMC Atrium Discovery .............................. 252
Using BMC Atrium Discovery ........................................... 254
Next steps .................................................................... 275
Security ...................................................................... 277
Security in BMC Atrium Discovery ..................................... 277
Automated discovery in secure environments ..................... 277
An appliance-based solution ............................................. 278
Appliance advantages ..................................................... 278
Software-based solutions ............................................... 279
Appliance hardening ....................................................... 279
User management .......................................................... 280
Appliance firewall .......................................................... 281
Windows proxy hardening .............................................. 281
Penetration testing .......................................................... 281
Understanding security audits .......................................... 282
Operating System Audit .................................................. 283
How to determine a false positive in an OS audit .................. 283
Information security ......................................................... 289
UI security ................................................................. 289
Credential vault security .................................................. 289
Sensitive data filters ......................................................... 290
System communications .................................................. 290
  Secure communications .................................................. 290
  Apache SSL key passphrases .......................................... 291
  End-user authentication .................................................. 291
  Secure export to CMDB .................................................. 291
  Ports used for System Communication ................................ 292
Discovery communications .............................................. 294
  Base device discovery .................................................. 294
  Mainframe: Ports used for discovery .................................. 298
  WBEM: Ports used for discovery ........................................ 298
  Ports required for extended discovery ................................ 298
  Dark space scanning ..................................................... 300
  RemQuery .................................................................. 302
Firewall Port Summary ..................................................... 303
  Scanning and standalone appliance ports .......................... 303
  Consolidation Appliance Ports ...................................... 305
  Windows proxy Ports ..................................................... 305
DISA Secure Technical Implementation Guidelines .................. 306
  Version compliance ....................................................... 306
  Changes to meet the Red Hat Enterprise Linux 6 and Apache 2.2 STIG in
  BMC Atrium Discovery 10.2 ............................................ 307
  STIG rules for RHEL6 addressed with restrictions .............. 307
  STIG rules for RHEL6 met using compliance script ............... 308
  STIG rules for RHEL6 not applicable to BMC Atrium Discovery 10.2 ... 313
  STIG rules for Apache addressed with restrictions ............... 319
  STIG rules for Apache not applicable to BMC Atrium Discovery 10.2 ... 319
Running in FIPS compliant mode ......................................... 321
  BMC Atrium Discovery and FIPS ...................................... 321
  Enabling NSS ............................................................. 323
PCI Data Security Standard compliance ............................... 330
  What is PCI DSS compliance? .......................................... 330
  Is BMC Atrium Discovery PCI DSS compliant? ..................... 330
Planning ............................................................... 331
Sizing and scalability considerations .................................. 331
Hardware requirements .................................................... 331
Installing BMC PATROL on a BMC Atrium Discovery system ........... 370

Upgrading .................................................. 373
Upgrading BMC Atrium Discovery .................................. 373
  Upgrade archive naming conventions .......................... 374
  Warnings .................................................. 374
  Upgrade considerations .................................... 375
  To run the upgrade ....................................... 376
  Post upgrade messages .................................... 378
  Resolving upgrade issues .................................. 381
Moving node keys ........................................... 385
  To move node keys in upgraded systems .................... 385
  To preserve IDs when using tw_model_wipe ................ 387
Configuring after upgrade .................................... 388

Using ....................................................... 389
Navigating the BMC Atrium Discovery UI ......................... 389
  Supported browsers ...................................... 389
  Page layout of the UI .................................... 390
  Tips for navigating the UI ................................ 396
  Downloading user documentation ............................ 396
Performing basic tasks ....................................... 396
  Setting preferences ...................................... 397
  Viewing the home page .................................... 398
  Viewing summary list pages ................................ 407
  Using View Object pages .................................. 417
Searching for data ........................................... 422
  Comparing the history of nodes ........................... 428
  Viewing dependency visualizations .......................... 434
  Managing attachments ..................................... 452
Configuring discovery ........................................ 455
  Configuring discovery settings .............................. 455
  Masking sensitive data .................................... 462
  Managing the discovery platform scripts ................... 464
  Adding privileged execution to commands .................. 469
  Setting up ports for OS fingerprinting ...................... 477
Running discovery .......................................... 478
  Viewing discovery status .................................. 478
Starting and stopping discovery ........................................... 482
Scanning IP addresses or ranges ......................................... 483
Excluding IP addresses and ranges from scanning .................. 488
Stalled discovery runs ..................................................... 490
Edge connectivity .......................................................... 492
Discovery targets ........................................................... 494
Credentials ................................................................. 494
Discovering UNIX, Linux and related OS based hosts .......... 498
Discovering Windows Hosts ............................................. 542
Discovering SNMP devices .............................................. 595
Discovering Load Balancers ............................................. 599
Discovering Management Controllers ................................. 602
Discovering ESX and ESXi hosts ....................................... 605
Discovering storage ...................................................... 619
Discovering database content .......................................... 626
Discovering WebLogic .................................................... 642
Discovering Tomcat ....................................................... 645
Discovering mainframe computers .................................... 647
Discovering IBMi .......................................................... 656
Discovering WebSphere .................................................. 657
Discovering IBM Power Systems ....................................... 659
Discovering Cisco Nexus switch IP addresses .............. 662
Improving discovery ...................................................... 662
Monitoring credential usage ............................................ 662
DiscoveryAccess page .................................................... 663
Discovery Conditions ..................................................... 670
Monitoring appliance performance .................................... 672
Appliance status .......................................................... 672
Engine performance ...................................................... 673
Pattern performance ..................................................... 674
Hardware statistics ....................................................... 676
Datastore cache performance ......................................... 678
DDD removal statistics .................................................. 680
Knowledge management ................................................ 682
Key terminology .......................................................... 682
To view the Knowledge management page ....................... 682
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing and editing a pattern module</td>
<td>684</td>
</tr>
<tr>
<td>Creating a pattern</td>
<td>686</td>
</tr>
<tr>
<td>Uploading knowledge</td>
<td>689</td>
</tr>
<tr>
<td>Downloading pattern modules</td>
<td>689</td>
</tr>
<tr>
<td>Configuring patterns in the UI</td>
<td>689</td>
</tr>
<tr>
<td>Executing patterns manually</td>
<td>690</td>
</tr>
<tr>
<td>Configuring pattern cleanup</td>
<td>693</td>
</tr>
<tr>
<td>Importing data</td>
<td>693</td>
</tr>
<tr>
<td>Importing network device data</td>
<td>693</td>
</tr>
<tr>
<td>Importing Hardware Reference Data</td>
<td>697</td>
</tr>
<tr>
<td>Importing CSV data</td>
<td>698</td>
</tr>
<tr>
<td>Managing your IT infrastructure</td>
<td>702</td>
</tr>
<tr>
<td>Viewing details of the IT infrastructure</td>
<td>702</td>
</tr>
<tr>
<td>Searching the IT infrastructure</td>
<td>703</td>
</tr>
<tr>
<td>Viewing a host</td>
<td>704</td>
</tr>
<tr>
<td>Viewing a cluster</td>
<td>721</td>
</tr>
<tr>
<td>Viewing a host container</td>
<td>721</td>
</tr>
<tr>
<td>Viewing a mainframe</td>
<td>724</td>
</tr>
<tr>
<td>Viewing a network device</td>
<td>729</td>
</tr>
<tr>
<td>Viewing load balancers</td>
<td>735</td>
</tr>
<tr>
<td>Viewing a network printer</td>
<td>743</td>
</tr>
<tr>
<td>Viewing an SNMP managed device</td>
<td>746</td>
</tr>
<tr>
<td>Viewing a subnet</td>
<td>759</td>
</tr>
<tr>
<td>Viewing a software instance</td>
<td>760</td>
</tr>
<tr>
<td>Viewing storage</td>
<td>761</td>
</tr>
<tr>
<td>Reporting on the infrastructure</td>
<td>773</td>
</tr>
<tr>
<td>Managing your business applications</td>
<td>773</td>
</tr>
<tr>
<td>Viewing details of business applications</td>
<td>774</td>
</tr>
<tr>
<td>Searching for application information</td>
<td>776</td>
</tr>
<tr>
<td>Viewing an application instance</td>
<td>777</td>
</tr>
<tr>
<td>Reporting on business applications</td>
<td>777</td>
</tr>
<tr>
<td>Auditing a data center</td>
<td>778</td>
</tr>
<tr>
<td>What is baselining</td>
<td>778</td>
</tr>
<tr>
<td>Automatic grouping</td>
<td>779</td>
</tr>
<tr>
<td>Manual grouping</td>
<td>784</td>
</tr>
<tr>
<td>Creating a host profile</td>
<td>794</td>
</tr>
</tbody>
</table>
Mapping your business applications ........................................ 795
  Creating SIs from discovered processes ................................ 795
Collaborative application mapping ...................................... 799
Creating SIs in a pattern ................................................... 844
Creating a static application model .................................... 845
Managing reporting ......................................................... 846
  Reporting basics .......................................................... 846
  Main reports page ....................................................... 847
  Application reports ..................................................... 855
  Infrastructure reports ................................................... 856
  Discovery reports ....................................................... 862
  Context-sensitive node reports ...................................... 864
Using the Search and Reporting service ................................. 873
  Using the Search service ............................................... 874
  Using Query Language .................................................. 875
  Context-sensitive reporting and linking .............................. 910
Exporting data ............................................................... 931
  Exporters ................................................................. 932
  Mapping files and sets .................................................. 932
  Adapters and adapter configurations ................................ 932
  To permit users to edit the adapter configuration ................. 933
Setting up and configuring an export .................................. 933
Running an export ......................................................... 950
Understanding the export process ..................................... 951
Export APIs ................................................................. 969
Integrating ................................................................. 976
CMDB synchronization ..................................................... 976
  Data model mapping .................................................... 978
  Supported CMDB versions ............................................. 978
  Deprecated versions of CMDB synchronization ..................... 979
Synchronization stages ................................................... 979
Resynchronization .......................................................... 979
Initial configuration ........................................................ 981
  To perform synchronization ........................................... 981
  To prevent synchronization at certain times ....................... 981
  To filter synchronization data ...................................... 981
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring dependency visualizations</td>
<td>1155</td>
</tr>
<tr>
<td>Configuring audit and application options</td>
<td>1161</td>
</tr>
<tr>
<td>Configuring model maintenance settings</td>
<td>1162</td>
</tr>
<tr>
<td>Configuring disk space monitor</td>
<td>1168</td>
</tr>
<tr>
<td>Managing disks and swap space</td>
<td>1170</td>
</tr>
<tr>
<td>Managing disks and swap space in a cluster</td>
<td>1173</td>
</tr>
<tr>
<td>Backing up and restoring the appliance</td>
<td>1177</td>
</tr>
<tr>
<td>Backing up and restoring a cluster</td>
<td>1181</td>
</tr>
<tr>
<td>Shutting down restarting and maintenance mode</td>
<td>1186</td>
</tr>
<tr>
<td>Shutting down restarting and maintenance in clusters</td>
<td>1188</td>
</tr>
<tr>
<td>Performing time synchronization</td>
<td>1189</td>
</tr>
<tr>
<td>Baseline configuration</td>
<td>1193</td>
</tr>
<tr>
<td>Managing standard data</td>
<td>1206</td>
</tr>
<tr>
<td>Setting up standard data categories</td>
<td>1206</td>
</tr>
<tr>
<td>Purging history and destroyed nodes</td>
<td>1213</td>
</tr>
<tr>
<td>Viewing the system taxonomy</td>
<td>1213</td>
</tr>
<tr>
<td>Managing clusters</td>
<td>1216</td>
</tr>
<tr>
<td>Information displayed on the Cluster Management page</td>
<td>1217</td>
</tr>
<tr>
<td>Cluster overview</td>
<td>1220</td>
</tr>
<tr>
<td>Fault tolerance</td>
<td>1221</td>
</tr>
<tr>
<td>Creating a cluster</td>
<td>1222</td>
</tr>
<tr>
<td>Adding a machine to an existing cluster</td>
<td>1225</td>
</tr>
<tr>
<td>Changing the machine that is the coordinator</td>
<td>1227</td>
</tr>
<tr>
<td>Removing a machine from a cluster</td>
<td>1230</td>
</tr>
<tr>
<td>Changing the address of a machine</td>
<td>1232</td>
</tr>
<tr>
<td>Reverting a cluster member into a standalone machine</td>
<td>1233</td>
</tr>
<tr>
<td>Shutting down and restarting machines in a cluster</td>
<td>1235</td>
</tr>
<tr>
<td>Shutting down and restarting clusters</td>
<td>1236</td>
</tr>
<tr>
<td>Consolidation</td>
<td>1237</td>
</tr>
<tr>
<td>Restrictions</td>
<td>1239</td>
</tr>
<tr>
<td>What is consolidated?</td>
<td>1239</td>
</tr>
<tr>
<td>Configuring consolidation</td>
<td>1240</td>
</tr>
<tr>
<td>Logging in to the appliance command line</td>
<td>1243</td>
</tr>
<tr>
<td>Default command line passwords</td>
<td>1243</td>
</tr>
<tr>
<td>The netadmin user</td>
<td>1245</td>
</tr>
<tr>
<td>The upload user</td>
<td>1253</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Using screen</td>
<td>1254</td>
</tr>
<tr>
<td>Using command line utilities</td>
<td>1258</td>
</tr>
<tr>
<td>Duplicate or enhanced functionality in user interface</td>
<td>1260</td>
</tr>
<tr>
<td>Common options in the utilities</td>
<td>1260</td>
</tr>
<tr>
<td>tw_scan_control</td>
<td>1261</td>
</tr>
<tr>
<td>tw_reasoningstatus</td>
<td>1269</td>
</tr>
<tr>
<td>tw_model_init</td>
<td>1270</td>
</tr>
<tr>
<td>tw_passwd</td>
<td>1270</td>
</tr>
<tr>
<td>tw_upduser</td>
<td>1271</td>
</tr>
<tr>
<td>tw_check_reports_model</td>
<td>1272</td>
</tr>
<tr>
<td>tw_adduser</td>
<td>1273</td>
</tr>
<tr>
<td>tw_cron_update</td>
<td>1274</td>
</tr>
<tr>
<td>tw_disco_export_platforms</td>
<td>1276</td>
</tr>
<tr>
<td>tw_restore</td>
<td>1277</td>
</tr>
<tr>
<td>tw_backup</td>
<td>1280</td>
</tr>
<tr>
<td>tw_listusers</td>
<td>1282</td>
</tr>
<tr>
<td>tw_imp_csv</td>
<td>1283</td>
</tr>
<tr>
<td>tw_imp_ciscoworks</td>
<td>1287</td>
</tr>
<tr>
<td>tw_disco_import_platforms</td>
<td>1289</td>
</tr>
<tr>
<td>tw_baseline</td>
<td>1289</td>
</tr>
<tr>
<td>tw_tripwire_rebaseline</td>
<td>1291</td>
</tr>
<tr>
<td>tw_disco_control</td>
<td>1292</td>
</tr>
<tr>
<td>tw_excluderanges</td>
<td>1293</td>
</tr>
<tr>
<td>tw_terminate_winproxy</td>
<td>1300</td>
</tr>
<tr>
<td>tw_deluser</td>
<td>1301</td>
</tr>
<tr>
<td>tw_convert_reports</td>
<td>1302</td>
</tr>
<tr>
<td>tw_tax_export</td>
<td>1303</td>
</tr>
<tr>
<td>tw_sign_winproxy_config</td>
<td>1304</td>
</tr>
<tr>
<td>tw_query</td>
<td>1305</td>
</tr>
<tr>
<td>tw_config_dashboards</td>
<td>1306</td>
</tr>
<tr>
<td>tw_sync_control</td>
<td>1307</td>
</tr>
<tr>
<td>tw_tax_import</td>
<td>1315</td>
</tr>
<tr>
<td>tw_ds_offline_compact</td>
<td>1317</td>
</tr>
<tr>
<td>tw_tax_deprecated</td>
<td>1320</td>
</tr>
<tr>
<td>tw_top</td>
<td>1321</td>
</tr>
<tr>
<td>tw_pattern_management</td>
<td>1322</td>
</tr>
</tbody>
</table>
Using TPL to enrich discovered data ........................................ 1619
VMware ESX and ESXi hosts discovery using custom patterns ...... 1623
Displaying custom attributes ................................................. 1631

**Troubleshooting** .............................................................. 1634
If you encounter a problem .................................................... 1634
  Viewing information about the appliance .............................. 1634
  Contacting Customer Support ............................................. 1636
Collecting diagnostics ....................................................... 1636
  Collecting additional data for support cases .................... 1636
Getting record data .............................................................. 1640
Contents of the logs ............................................................. 1642
Changing log levels at runtime .............................................. 1647

**Frequently Asked Questions** ........................................... 1648
What is the performance impact of running BMC Atrium Discovery? .. 1648
What can BMC Atrium Discovery discover in the network? .......... 1648
What discovery techniques do you use? ............................... 1648
Will any network security need to be disabled for the discovery process? 1648
What is the impact of my applications running on platforms that are not supported by BMC Atrium Discovery? ..................... 1649
Can the product introduce any risk into my network or application infrastructure? ......................................................... 1649
Do I need to install any software on other computers? ............ 1649
Why is agent-free discovery so important? ........................... 1649
How do I reset the BMC Atrium Discovery user password? ....... 1649

**Troubleshooting clusters** .................................................. 1650
My cluster will not start! ...................................................... 1650
To determine the health of a cluster ..................................... 1651
Removing all failed machines from the cluster ...................... 1653
A machine has failed in my non-fault tolerant cluster ............. 1654
Restoring a failed and forcibly removed coordinator ............... 1654

**Troubleshooting Windows proxies** .................................... 1655
Checking the keys and certificates ........................................ 1655
Checking for an omniORB.cfg file ........................................ 1655

Writing efficient regular expressions .................................... 1656
Why is regular expression efficiency important? .................... 1656
Anatomy of an inefficient regular expression ......................... 1656
Guidelines for writing efficient regular expressions .................. 1659
Regular expression optimisations for TPL triggers ...................... 1661
Further Reading ................................................................. 1663
Useful tools ................................................................. 1663
Diagnosing Hostname Problems ............................................. 1664
  Visualizations and export do not display ............................... 1664
  Cannot access the UI — 500 internal server error .................... 1664
Unmounting partitions on used disks ....................................... 1664
Appliance does not have an IP on eth0 .................................... 1666
  eth0 is present but does not have an IP address .................... 1666
  eth1 has an IP address instead of eth0 ................................ 1666
Support information ....................................................... 1666
  Contacting Customer Support ........................................... 1666
  Support status .......................................................... 1666
Best practices ............................................................... 1667
Overview ................................................................. 1667
Security ................................................................. 1667
Planning ................................................................. 1668
Installing ............................................................... 1668
Integrating .............................................................. 1668
Maintaining ............................................................. 1669
Index ................................................................. 1670
This is the home page for BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) 10.2 documentation. The documentation is divided into the following sections:

- **Release notes and notices** — Accompany each product release and provide details of new features introduced in that release as well as known issues/workarounds. For information on between-release updates or EFix releases, see the readme file that is provided with the update.
- **Getting started** — Provides an introduction to BMC Atrium Discovery. It is intended for any staff who are unfamiliar with BMC Atrium Discovery.
- **Security** — Provides an overview of the security aspects of BMC Atrium Discovery.
- **Planning** — Provides information that will be useful when planning a BMC Atrium Discovery deployment.
- **Installing** — Provides detailed pre-requisites and procedures for installing BMC Atrium Discovery on hardware and virtual machines.
- **Upgrading** — Provides detailed pre-requisites and procedures for upgrading BMC Atrium Discovery.
- **Using** — Describes the day to day use of the product. It is intended for IT operations staff and managers who are using and managing BMC Atrium Discovery, including viewing, creating, exporting, searching, and reporting on data from an infrastructure or a business perspective. Information on the Using the Search and Reporting service is also provided.
- **Integrating** — Describes integrating other products with BMC Atrium Discovery.
- **Administering** — Describes how to set up and configure the product. It is intended for IT operations staff who are responsible for setting up, managing and configuring BMC Atrium Discovery and any associated Windows proxies. It describes options on the BMC Atrium Discovery user interface that are only available to users with administration privileges as well as command line access to certain parts of the product which are not accessible through the UI.
- **Developing** — Provides information on the Pattern Language (TPL) which is used to describe applications, products and other real-world entities that have been modeled in BMC Atrium Discovery. Also contains information to help you understand the BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) node lifecycle.
- **Troubleshooting** — Provides information on possible problems that you might encounter, troubleshooting and debugging tips, and how to collect data that Customer Support might need to solve problems.
Updated SSH server

The SSH server installed on the BMC Atrium Discovery 10.2 appliance has been modified. This modification removes some weak ciphers and HMAC algorithms (and non-FIPS approved) from the list of allowed connections. In practice this should not affect most users, however, some older versions of ssh clients will be unable to connect.

The following versions, or later, of the following clients have been tested and are known to work:

- PuTTY (0.64)
- WinSCP (5.7.2)
- mRemoteNG (1.72 using ssh2)
- MobaXterm (6.5)
- KiTTY (0.63.2.2)
- Bitvise SSH Client (6.22)
Release notes and notices

This set of Release Notes describes versions 10.2, 10.2.0.1, and 10.2.0.2 of BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery).

Version history

BMC Atrium Discovery version 10.2 was released on 28 May 2015.

BMC Atrium Discovery version 10.2.0.1 was released on 10 August 2015.

BMC Atrium Discovery version 10.2.0.2 was released on 30 September 2015.

10.2.0.2 is a patched version of 10.2 which supersedes version 10.2 and 10.2.0.1. You can only upgrade to version 10.2.0.2 from version 10.1 and later.

BMC Atrium Discovery version 10.2.0.3 was released on 13 October 2015.

10.2.0.3 is a patched version of 10.2 which supersedes versions 10.2, 10.2.0.1 and 10.2.0.2. You can only upgrade to version 10.2.0.3 from version 10.1 and later.

Release Notes

These Release Notes detail the following information:

- Flash
- 10.2 Enhancements
- Known and corrected issues
- Important information for users of BMC Atrium CMDB
- Technical bulletins
- Product announcements
- Limitations and restrictions of this version
Updated SSH server

The SSH server installed on the BMC Atrium Discovery 10.2 appliance has been modified. This modification removes some weak ciphers and HMAC algorithms (and non-FIPS approved) from the list of allowed connections. In practice this should not affect most users, however, some older versions of ssh clients will be unable to connect.

The following versions, or later, of the following clients have been tested and are known to work:

- PuTTY (0.64)
- WinSCP (5.7.2)
- mRemoteNG (1.72 using ssh2)
- MobaXterm (6.5)
- KITTY (0.63.2.2)
- Bitvise SSH Client (6.22)

BMC Atrium Discovery OS and application software have been tested and found to be free from susceptibility to date/time issues which may occur as the result of the additional day in a leap year.

Flash

10.2 Enhancements

This section describes the new features and enhancements that have been introduced in the release of BMC Atrium Discovery version 10.2, 10.2.0.1, 10.2.0.2, and 10.2.0.3.

If you are new to BMC Atrium Discovery, BMC recommends that you refer to Getting started as an introduction to using the product.

You can only upgrade to 10.2.0.3 from 10.1 and later. The upgrade procedure is described in Upgrading BMC Atrium Discovery.
No architecture changes required

No architecture changes required

When upgrading from 9.x to 10.1 and from there to 10.2 there is no need to change your deployment architecture. Rather, BMC strongly recommend that you upgrade using your existing architecture and test version 10.x in a known configuration. If testing reveals that you require the performance improvements offered by a cluster, you can then add hardware and replace one or more version 10.x machines with a cluster.

Updated SSH server

The SSH server installed on the BMC Atrium Discovery 10.2 appliance has been modified. This modification removes some weak ciphers and HMAC algorithms (and non-FIPS approved) from the list of allowed connections. In practice this should not affect most users, however, some older versions of ssh clients will be unable to connect.

The following versions, or later, of the following clients have been tested and are known to work:

- PuTTY (0.64)
- WinSCP (5.7.2)
- mRemoteNG (1.72 using ssh2)
- MobaXterm (6.5)
- KiTTY (0.63.2.2)
- Bitvise SSH Client (6.22)

What's new in BMC Atrium Discovery version 10.2?

BMC Atrium Discovery 10.2.0.3

The following new features and enhancements have been introduced in BMC Atrium Discovery version 10.2.0.3. BMC Atrium Discovery version 10.2.0.3 was released on 13 October 2015.

BMC Atrium Discovery 10.2.0.3 is a patch release to BMC Atrium Discovery version 10.2 resolving a number of defects. For more information see Defects resolved in this version.

The BMC Atrium Discovery 10.2.0.3 release adds support for HP iLO versions 3 and 4.
The release also introduces an upgraded base kickstart DVD with an updated kernel and device driver set. This enables you to install BMC Atrium Discovery 10.2.0.3 on new hardware supported by Red Hat Enterprise Linux (RHEL) 6.7. The previous base OS used was RHEL 6.3, but the device driver set included in the kickstart did not recognize later hardware at install. Whether you install from a kickstart DVD or upgrade from a previous version, BMC Atrium Discovery 10.2.0.3 runs on RHEL 6.7.

**BMC Atrium Discovery 10.2.0.2**

The following new features and enhancements have been introduced in BMC Atrium Discovery version 10.2.0.2. BMC Atrium Discovery version 10.2.0.2 was released on 30 September 2015.

BMC Atrium Discovery 10.2.0.2 is a patch release to BMC Atrium Discovery version 10.2 resolving a number of defects. For more information see [Defects resolved in this version](#).

Improvements have been made to the CMDB synchronization.

- CMDB resync is now multi-threaded. You can specify the number of threads to use, though increasing from the default puts more load on the CMDB system. See the Resync Threads option in [Setting up a CMDB synchronization connection](#).
- The CMDB sync connection batch size is now configurable. You can specify the batch size to prevent timeouts during sync and resync batched operations. See the Batch Size option in [Setting up a CMDB synchronization connection](#).
- You can now upgrade even if the CMDB sync queue is not empty.
- During a resync the UI shows counts of all the actions to be performed. This information is now logged.

If you are seeing poor delete performance in CMDB sync, please read "Are you experiencing ADDM CMDB Sync Time out issue? Read this post." and contact customer support for details.

**BMC Atrium Discovery 10.2.0.1**

The following new features and enhancements have been introduced in BMC Atrium Discovery version 10.2.0.1. BMC Atrium Discovery version 10.2.0.1 was released on 10 August 2015.

BMC Atrium Discovery 10.2.0.1 is a patch release to BMC Atrium Discovery version 10.2 resolving a number of defects. For more information see [Defects resolved in this version](#).

Changes have been made to the workflow for configuring multitenancy in this release. For more information see [multitenancy](#) and [setting up a CMDB synchronization connection](#).

BMC Atrium Discovery version 10.2 was released on 28 May 2015.
CMDB Synchronization improvements

In previous releases, at each synchronization, BMC Atrium Discovery had to query the CMDB to determine the current state, and this was where most of the time was spent in synchronization. From this release BMC Atrium Discovery maintains an authoritative model of its view of the world in the datastore, the shadow copy. From there it is quicker and simpler to determine the changes that need to be sent to the CMDB. Typically data centers do not change very much between scans, so it is just these changes that are sent to CMDB, and this can be done very quickly.

Occasionally the model stored in the CMDB dataset can become out of step with the shadow copy for a CMDB connection and require resynchronization. For example, if CMDB tools have been used to modify the dataset. BMC Atrium Discovery will recommend a resynchronization if it detects errors of this kind. A tool is provided in the UI which resynchronizes the CMDB with BMC Atrium Discovery.

The CMDB sync feature now supports connections to multiple CMDBs.

The filtering UIs have been improved to make them more intuitive, and you can preview the synchronization results for a set of items using CMDB Sync Preview.

A command line tool, tw_sync_control, is provided so that synchronization operations can be scripted.

The following TPL features have been added to the syncmapping block in TPL:

- **overrides** directive
- **Impacted** and **ImpactWeight**

Blade enclosure discovery

This release supports the discovery of blade enclosures and relating enclosures to contained blades. The blade chassis is modeled as a Host Container node and the blades are modeled as Host nodes.

At the time of the release of BMC Atrium Discovery 10.2, we support:

- Cisco UCS
- Dell M1000e
- HP BladeSystem c7000
- IBM BladeCenter
- Sun Blade 6000 system

Management controller discovery

In addition to that already mentioned in blade enclosure discovery, this release introduces support for management controller discovery. These are modeled as Management Controller nodes.
At the time of the release of BMC Atrium Discovery 10.2, we support:

- Dell DRACs
- Dell CMC
- IBM RSA
- SUN ILOM and Oracle (Sun) ILOMs
- HP iLO
- Cisco IMC

A new credential type for the XML API, is provided to discover some management controllers. At the time of release, HP iLO and Cisco IMC are discovered using this credential type. The other management controllers listed are discovered using SNMP.

The HP Integrity iLO devices are not discovered at the time of release.

Where subsequent TKU releases increase coverage of management controller types discovered, this will be documented in Configipedia.

**Host/Storage linking**

In this release we have made improvements in linking hosts to the storage that they are actually using on storage arrays or local disks.

**Local drives**

Where a host has a file system on a local disks we now link the FileSystem to the DiskDrive. On Linux hosts, BMC Atrium Discovery expands the logical volume managers and/or software RAID configuration to find the actual physical drives and link the DiskDrive nodes representing them to the FileSystem. In this way a single FileSystem can be linked to multiple physical disks. The software RAID configuration is not modeled. Physical drives in these cases means physical or virtual (for a VM) rather than logical volumes.

Linux can mount drives using UUIDs, these can now be linked to the physical disk drives.

The DiskDrive nodes are also linked to the Host node to cater for disks without file systems.

**Storage**

For storage arrays we now link FileSystems to the StorageVolume (LUN) which provides the storage. On Linux we support multipath as well as simple (single path) access. We also link the StorageVolume nodes to the Host. This enables you to identify LUNs which the host can use but is not currently using. This may be misconfiguration, or an opportunity for expansion.

**VMware ESX**

VMware ESX data stores (modeled as file systems in BMC Atrium Discovery) have been discovered in previous releases, they are now linked to the storage volumes on which they consume storage, or the local DiskDrive. This uses the vSphere API so is not supported though ssh discovery of older ESX hosts.
On Linux VMs, Raw Disk Maps appear as a SAN LUN.

Storage mounted as a SCSI LUN is now linked. This enables you to see all storage volumes that the host has mounted, even if they are unused and there is no file system. It provides information on the volume itself, for example the visible capacity.

**SAN Switch discovery**

Discovery of SAN switches is now supported for Brocade and Cisco. The basic SNMP identification of SAN switches is available to all customers. For customers with the Storage TKU, retrieval of Fibre Channel Ports information is supported.

**Miscellaneous**

The following miscellaneous features have been added:

**Changes in the Extended RDB adapter mapping set**

In the Extended RDB adapter mapping set, the following fields are no longer exported with the Network Interface CI information by default:

- interface_name
- interface_speed
- interface_duplex
- interface_negotiation

To export the mentioned details, you could customize the Extended RDB adapter mapping set by uploading these `host.xml` and `networkinterface.xml` files. For more information on customizing a mapping set, see Managing mapping sets.

When importing data to the external database, you need to modify the default database scripts used for initial configuration on the RDB side to match the introduced changes in the export schema. You can download a sample customized scripts for Oracle, MySQL, and for SQLServer: `SQLServer.sql` and `SQLServer.osql`.

**New TPL function**

A new TPL function `binary.toValue` has been added which enables you to convert a given binary value into a specified format. The binary value is typically a value retrieved via SNMP using the binary settings on `discovery.snmpGet` or `discovery.snmpGetTable`.

**Model changes**

The changes in the BMC Atrium Discovery content model between versions 10.1 and 10.2 are described in this section:

- **Nodes added**
- **Nodes removed**
• **Nodes changed**

**Nodes added**

- HardwareDetail
- ManagementController

**Nodes removed**

The following node kinds have been removed in BMC Atrium Discovery version 10.2:

- **None**

**Nodes changed**

**DeviceInfo**

**New attributes**

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform</td>
<td>platform string</td>
<td>Platform scripts used in discovery.</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>firmware_version string</td>
<td>Firmware version.</td>
</tr>
</tbody>
</table>

**New relationships**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Device</td>
<td>DeviceInfo: Primary: Inference: InferredElement: StorageDevice</td>
<td>Inferred Storage Device.</td>
</tr>
<tr>
<td>Printer</td>
<td>DeviceInfo: Primary: Inference: InferredElement: Printer</td>
<td>Inferred Printer.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>DeviceInfo: Primary: Inference: InferredElement: SNMPManagedDevice</td>
<td>Inferred SNMP Managed Device.</td>
</tr>
</tbody>
</table>

**DiscoveredCard**

**Changed attribute**

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slot Number</td>
<td>slot_number int</td>
<td>Slot Number.</td>
</tr>
</tbody>
</table>

Is changed to:
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slot</td>
<td>slot string</td>
<td>Slot.</td>
</tr>
</tbody>
</table>

### DiscoveryAccess

#### New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized Management Controller</td>
<td>DiscoveryAccess: AccessOptimization: InferredElement: ManagementController</td>
<td>Management Controller corresponding to an access that was optimized away.</td>
</tr>
<tr>
<td>Management Controller</td>
<td>DiscoveryAccess: Associate: Inference: InferredElement: ManagementController</td>
<td>Management Controller inferred from this access.</td>
</tr>
</tbody>
</table>

### DiskDrive

#### New attributes

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current State</td>
<td>current_state string</td>
<td>Current state of the disk drive.</td>
</tr>
<tr>
<td>Virtual</td>
<td>virtual boolean</td>
<td>Virtual flag.</td>
</tr>
</tbody>
</table>

#### New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>DiskDrive: ContainedDrive: Containment: Container: Host</td>
<td>Host for this disk.</td>
</tr>
<tr>
<td>Physical Disks</td>
<td>DiskDrive: Consumer: StorageUse: Provider: DiskDrive</td>
<td>Physical disk for this virtual disk.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>DiskDrive: Provider: StorageUse: Consumer: DiskDrive</td>
<td>Virtual disks using this physical disk.</td>
</tr>
<tr>
<td>Physical Volumes</td>
<td>DiskDrive: Consumer: StorageUse: Provider: StorageVolume</td>
<td>Physical storage volume for this virtual disk.</td>
</tr>
</tbody>
</table>
Endpoint
This is an internal implementation detail. Do not attempt to create or alter.
New relationship

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Controllers</td>
<td>Endpoint: Endpoint: ChosenEndpoint: Device: ManagementController</td>
<td>Management Controllers discovered using this endpoint.</td>
</tr>
</tbody>
</table>

FibreChannelPort
New relationship

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Device</td>
<td>FibreChannelPort: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: NetworkDevice</td>
<td>Network device for this port.</td>
</tr>
</tbody>
</table>

FileSystem
New attributes

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Has Qtrees has_qtrees boolean</td>
<td>Has related Qtrees.</td>
</tr>
<tr>
<td></td>
<td>Is a Qtree qtree boolean</td>
<td>Is a Qtree.</td>
</tr>
<tr>
<td></td>
<td>Owning Volume root_volume string</td>
<td>Root Volume.</td>
</tr>
</tbody>
</table>

New relationship

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Drive</td>
<td>FileSystem: Consumer: StorageUse: Provider: DiskDrive</td>
<td>DiskDrive providing storage.</td>
</tr>
</tbody>
</table>

Host
New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Drives</td>
<td>Host: Container: Containment: ContainedDrive: DiskDrive</td>
<td>DiskDrives contained in this Host.</td>
</tr>
<tr>
<td>Disk Drives</td>
<td>Host: Consumer: StorageUse: Provider: DiskDrive</td>
<td>DiskDrives used by this Host.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>Storage Volumes</td>
<td>Host: Consumer: StorageUse: Provider: StorageVolume</td>
<td>Storage Volumes used by this Host.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: ManagementController</td>
<td>Next Management Controller Identity.</td>
</tr>
<tr>
<td>Managed by Management Controller</td>
<td>Host: ManagedElement: Management: Manager: ManagementController</td>
<td>Management Controller that manages this Host.</td>
</tr>
</tbody>
</table>

HostContainer

New attribute

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not shown in UI</td>
<td>asset_tag string</td>
<td>Dell Asset tag.</td>
</tr>
</tbody>
</table>

New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host Members</td>
<td>HostContainer: HostContainer: HostContainment: ContainedHost: Host</td>
<td>Hosts that are within this container.</td>
</tr>
<tr>
<td>Network Device Members</td>
<td>HostContainer: HostContainer: HostContainment: ContainedHost: NetworkDevice</td>
<td>Network Devices that are within this container.</td>
</tr>
<tr>
<td>SNMP Managed Device Members</td>
<td>HostContainer: HostContainer: HostContainment: ContainedHost: SNMPManagedDevice</td>
<td>SNMP Managed Devices that are within this container.</td>
</tr>
<tr>
<td>Management Controller Members</td>
<td>HostContainer: HostContainer: HostContainment: ContainedHost: ManagementController</td>
<td>Management Controllers that are within this container.</td>
</tr>
<tr>
<td>Managed by Network Device</td>
<td>HostContainer: ManagedElement: Management: Manager: NetworkDevice</td>
<td>Network Device that manages this container.</td>
</tr>
<tr>
<td>Managed by Management Controller</td>
<td>HostContainer: ManagedElement: Management: Manager: ManagementController</td>
<td>Management Controller that manages this container.</td>
</tr>
<tr>
<td>Details</td>
<td>HostContainer: ElementWithDetail: Detail: Detail: HardwareDetail</td>
<td>Hardware Details for this container.</td>
</tr>
</tbody>
</table>

IPAddress:

New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Controller</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: ManagementController</td>
<td>The Management Controller that this IPv4 address belongs to.</td>
</tr>
</tbody>
</table>
### Location

**New relationship**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Controllers in this Location</td>
<td>Location: Location: ElementInLocation: ManagementController</td>
<td>Management Controllers at this location.</td>
</tr>
</tbody>
</table>

### Mainframe

**New relationships**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
</tbody>
</table>

### NetworkDevice

**New relationship**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fibre Channel Ports</td>
<td>NetworkDevice:DeviceWithInterface: DeviceInterface: InterfaceOfDevice:FibreChannelPort</td>
<td>FC ports of this NetworkDevice.</td>
</tr>
</tbody>
</table>

### NetworkInterface

**New relationship**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Controller</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: ManagementController</td>
<td>The Management Controller that this network interface belongs to.</td>
</tr>
</tbody>
</table>

### OrganisationalUnit

**New relationship**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Controllers in this OU</td>
<td>OrganisationalUnit: Owner: Ownership: OwnedItem: ManagementController</td>
<td>Management Controllers in this unit.</td>
</tr>
</tbody>
</table>
**Printer**

New attribute

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firmware Version</td>
<td>firmware_version string</td>
<td>Firmware version.</td>
</tr>
</tbody>
</table>

New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
</tbody>
</table>

**SNMPManagedDevice**

New attribute

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firmware Version</td>
<td>firmware_version string</td>
<td>Firmware version.</td>
</tr>
</tbody>
</table>

New relationships

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Container</td>
<td>SNMPManagedDevice: ContainedHost: HostContainment: HostContainer: HostContainer</td>
<td>Host container of which this is a member.</td>
</tr>
</tbody>
</table>

**StorageDevice**

New attribute
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firmware Version</td>
<td>firmware_version string</td>
<td>Firmware version.</td>
</tr>
</tbody>
</table>

**New relationships**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: ManagementController}</td>
<td>Previous Management Controller Identity.</td>
</tr>
</tbody>
</table>

**StorageVolume**

**New attribute**

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Blocks</td>
<td>num_blocks int</td>
<td>Number of blocks.</td>
</tr>
</tbody>
</table>

**New relationships**

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Volume Providers</td>
<td>StorageVolume: Consumer: StorageUse: Provider: StorageVolume</td>
<td>Storage volumes providing this Volume.</td>
</tr>
<tr>
<td>Storage Volume Consumers</td>
<td>StorageVolume: Provider: StorageUse: Consumer: StorageVolume</td>
<td>Storage volumes consuming this Volume.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>StorageVolume: Provider: StorageUse: Consumer: DiskDrive</td>
<td>Virtual disks consuming this Volume.</td>
</tr>
</tbody>
</table>

**Changes to Discovery Commands**

The following sections show the discovery command changes between BMC Atrium Discovery versions.

The following changes are not shown:

- Entirely new discovery platforms
- Changes to comments only
- Commands which have been removed and not replaced
- Changes to echo only statements
Discovery command changes from 10.1 to 10.2

Click here to show the discovery command changes from 10.1 to 10.2

Solaris

**getHostInfo**

The following code:

```bash
platdir=/usr/platform/`uname -m`/sbin/sparcv9
elif [ -x /usr/platform/`uname -i`/sbin/prtdiag ]; then
    platdir=/usr/platform/`uname -i`/sbin
else
    platdir=/sbin
fi
```

is replaced with:

```bash
platdir=/usr/platform/`uname -m`/sbin/sparcv9
elif [ -x /usr/platform/`uname -i`/sbin/prtdiag ]; then
    platdir=/usr/platform/`uname -i`/sbin
elif [ -x /usr/sbin/prtdiag ]; then
    platdir=/usr/sbin
else
    platdir=/sbin
fi
```

**getNetworkInterfaces**

The following code:

```bash
if [ $GOT_SPEED_DUPLEX -eq 1 ]; then # ndd -get may provide negotiation info
    for NAME in $LINKNAMES; do
        NIC_TYPE=`echo $NAME | sed 's/[0-9.]*//g'`
        case $NIC_TYPE in
            dmfe | bge | nxge | igb)
                echo 'NDD :' $NAME ':adv_autoneg_cap:' `PRIV_NDD $NDD -get /dev/$NAME
```

is replaced with:

```bash
if [ $GOT_SPEED_DUPLEX -eq 1 ]; then # ndd -get may provide negotiation info
    for NAME in $LINKNAMES; do
        NIC_TYPE=`echo $NAME | sed 's/[0-9.]*//g'`
```
case $NIC_TYPE in
  dmfe | bge | nxge | igb)
    echo 'NDD :' $NAME ':adv_autoneg_cap:' `PRIV_NDD $NDD -get /dev/$NAME
The following code:

```bash
vars_fjgq="link_mode link_speed adv_autoneg_cap"
vars_fjgi="link_mode link_speed adv_autoneg_cap"
vars_igb="link_duplex link_speed adv_autoneg_cap"
LISTOFTYPES=`echo "$LINKNAMES" | sed 's/[0-9.]*$//g' | sort -u`
LISTOFSETTYPES=`echo $LISTOFTYPES | sed -e 's/bge//' -e 's/dmfe//' -e 's/igb//'`
for iface in $LISTOFSETTYPES; do
  eval initial_$iface=`PRIV_NDD $NDD -get /dev/$iface instance 2>/dev/null`
done
for NAME in $LINKNAMES; do
  NIC_TYPE=`echo $NAME | sed 's/[0-9.]*//g'`
  NIC_NUMBER=`echo $NAME | sed 's/[a-z]*//'`
  eval vars=$vars_$NIC_TYPE
  case $NIC_TYPE in

is replaced with:

```bash
vars_fjgq="link_mode link_speed adv_autoneg_cap"
vars_fjgi="link_mode link_speed adv_autoneg_cap"
vars_igb="link_duplex link_speed adv_autoneg_cap"
LISTOFTYPES=`echo "$LINKNAMES" | sed 's/[0-9.]*$//g' | sort -u`
LISTOFSETTYPES=`echo $LISTOFTYPES | sed -e 's/bge//' -e 's/dmfe//' -e 's/igb//'`
for iface in $LISTOFSETTYPES; do
  eval initial_$iface=`PRIV_NDD $NDD -get /dev/$iface instance 2>/dev/null`
done
for NAME in $LINKNAMES; do
  NIC_TYPE=`echo $NAME | sed 's/[0-9.]*//g'`
  NIC_NUMBER=`echo $NAME | sed 's/[a-z]*//'`
  eval vars=$vars_$NIC_TYPE
  case $NIC_TYPE in

Linux

`getHostInfo`

The following code is added:

```bash
# The specification only allows for the physical RAM size to be given in either
# KB or MB.  [System Management BIOS (SMBIOS) Reference Specification, p.86 -
# http://www.dmtf.org/sites/default/files/standards/documents/DSP0134_2.8.0.pdf]`
if [ -f /usr/sbin/dmidecode ]; then
    dmidecode_ram=`PRIV_DMIDECODE /usr/sbin/dmidecode -t 17 2>/dev/null | awk '{
        if ($3 == "kB" || $3 == "KB")
            size += $2
        else if ($3 == "MB")
            size += $2*1024
    }
    END {
        print size
    }'`
    if [ "${dmidecode_ram}" != "" ]; then
        ram="${dmidecode_ram}KB"
    fi
fi

dmidecode

The following code is added:

```bash
# ss requires superuser privileges to display process identifiers (PIDs)
# for ports opened by processes not running as the current user
PRIV_SS() {
    "$@
}
```

getDeviceInfo

The following code:

```bash
ihn=`hostname 2>/dev/null`
if [ "$ihn" = "localhost" ]; then
    ihn=`hostname 2>/dev/null`
fi
echo 'hostname:' $ihn
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" ]; then
    dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
    if [ "$dns_domain" = "" ]; then
        dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
    fi
fi
```

is replaced with:

```bash
ihn=`hostname 2>/dev/null`
echo 'hostname:' `hostname 2>/dev/null`
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" ]; then
    dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
    if [ "$dns_domain" = "" ]; then
        dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
    fi
fi
```

initialise

The following code is added:

```bash
# ss requires superuser privileges to display process identifiers (PIDs)
# for ports opened by processes not running as the current user
PRIV_SS() {
    "$@
}
```
VMware ESX

gETCHOSTINFO

The following code is added:

```bash
# The specification only allows for the physical RAM size to be given in either
# KB or MB. [System Management BIOS (SMBIOS) Reference Specification, p.86 -
# http://www.dmtf.org/sites/default/files/standards/documents/DSP0134_2.8.0.pdf]
if [ -f /usr/sbin/dmidecode ]; then
    dmidecode_ram=`PRIV_DMIDECODE /usr/sbin/dmidecode -t 17 2>/dev/null | awk '
        /Size:/ {
            if ($3 == "kB" || $3 == "KB")
                size += $2
            else if ($3 == "MB")
                size += $2*1024
        }'
    END {
        print size
    }'
    if [ "${dmidecode_ram}" != "" ]; then
        ram="${dmidecode_ram}KB"
    fi
fi
```

initialise

The following code is added:

```bash
# ss requires superuser privileges to display process identifiers (PIDs)
# for ports opened by processes not running as the current user
PRIV_SS() {
    "@*
}
```

GETDEVICEINFO

The following code:

```bash
ihn=`hostname 2>/dev/null`
if [ "$ihn" = "localhost" ]; then
    ihn=`hostname 2>/dev/null`
fi
echo 'hostname:' $ihn
```
is replaced with:

```
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
DNS_DOMAIN=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$DNS_DOMAIN" = '' -a -r /etc/resolv.conf ]; then
```

---

**Discovery command changes from 10.0 to 10.1**

Click here to show the discovery command changes from 10.0 to 10.1

---

**Solaris**

**getHostInfo**

The following code:

```
eelif [ -x /usr/platform/`uname -m`/sbin/sparcv9/prtdiag ]; then
  platdir=/usr/platform/`uname -m`/sbin/sparcv9
else
  platdir=/usr/platform/`uname -i`/sbin
```

is replaced with:

```
eelif [ -x /usr/platform/`uname -m`/sbin/sparcv9/prtdiag ]; then
  platdir=/usr/platform/`uname -m`/sbin/sparcv9
  elif [ -x /usr/platform/`uname -i`/sbin/prtdiag ]; then
    platdir=/usr/platform/`uname -i`/sbin
else
  platdir=/sbin
```

---

The following code is added:

```
# Solaris 9 and earlier do not support zones directly.
# There isn't any commands to run inside a branded zone to detect if it's a zone.
# However, the zonename is usually listed in /etc/mnttab (for all solaris zones)
# For example:
# Solaris 11: mnttab /zones/sol11z1/root/etc/mnttab mntfs nodevices,zone=sol11z1,sharezone
```
AIX

getHostInfo

The following code:

```bash
echo 'begin lparstat:'
cat /tmp/tideway.$$
echo 'end lparstat'
```

is replaced with:

```bash
echo 'begin lparstat:'
cat /tmp/tideway.$$  # Adding extra echo to workaround the AIX v7.1 bug IV56200
# which description is "LINE FEED MISSING IN LPARSTAT -I COMMAND OUTPUT"
echo 'end lparstat'
```

getDeviceInfo

The following code:

```bash
echo 'os:' $os
maintlevel=`oslevel -r 2>/dev/null`  
if [ $? -ne 0 ]; then
  maintlevel=""
fi
```

is replaced with:

```bash
echo 'os:' $os
maintlevel=`oslevel -r 2>/dev/null`  
if [ $? -ne 0 ]; then
  maintlevel=""
fi
```
```bash
echo 'os:' $os
maintlevel=`oslevel -s 2>/dev/null`
if [ $? -ne 0 ]; then
    maintlevel=""
fi
```

**Linux**

getDeviceInfo

The following code:

```bash
os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed -e 's/^[ ]//']
if [ "$os" = "(none)" ]; then
    os=""
else
    # Check to see if its a variant of Red Hat
    rpm -q oracle-logos > /dev/null 2>&1
    if [ $? -eq 0 ]; then
```

is replaced with:

```bash
os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed -e 's/^[ ]//']
if [ "$os" = "(none)" ]; then
    os=""
elif [ "$os" != "" ]; then
    # Check to see if its a variant of Red Hat
    rpm -q oracle-logos > /dev/null 2>&1
    if [ $? -eq 0 ]; then
```

**VMware ESX**

getDeviceInfo

The following code:

```bash
os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed -e 's/^[ ]//']
if [ "$os" = "(none)" ]; then
    os=""
else
    # Check to see if its a variant of Red Hat
    rpm -q oracle-logos > /dev/null 2>&1
    if [ $? -eq 0 ]; then
```
Discovery command changes from 9.0 SP2 to 10.0

**Solaris**

getHostInfo

The following code:

```bash
if [ $run_prtdiag -eq 1 ]; then
    if [ -x /usr/platform/`uname -m`/sbin/prtdiag ]; then
        platdir=/usr/platform/`uname -m`/sbin
    else
        platdir=/usr/platform/`uname -i`/sbin
    fi
```

is replaced with:

```bash
if [ $run_prtdiag -eq 1 ]; then
    if [ -x /usr/platform/`uname -m`/sbin/prtdiag ]; then
        platdir=/usr/platform/`uname -m`/sbin
    elif [ -x /usr/platform/`uname -m`/sbin/sparcv9/prtdiag ]; then
        platdir=/usr/platform/`uname -m`/sbin/sparcv9
    else
        platdir=/usr/platform/`uname -i`/sbin
    fi
```

getNetworkInterfaces

The following code:

```bash
    case $NIC_TYPE in
        dmfe | bge | nxge)
            echo 'NDD :' $NAME ':adv_autoneg_cap:' `PRIV_NDD $NDD -get /dev/$NAME adv_autoneg_cap 2>/dev/null`
```

is replaced with:

```bash
case $NIC_TYPE in
dmfe | bge | nxge | igb)
  echo 'NDD :' $NAME ':adv_autoneg_cap:' `PRIV_NDD $NDD -get /dev/$NAME
```

The following code:

```bash
vars_fjqe="link_mode link_speed adv_autoneg_cap"
vars_fjgi="link_mode link_speed adv_autoneg_cap"
LISTOFTYPES=`echo "$LINKNAMES" | sed 's/[0-9]*$//g' | sort -u`
LISTOFSETTYPES=`echo $LISTOFTYPES | sed -e 's/bge//' -e 's/dmfe//'`
for iface in $LISTOFSETTYPES; do
eval initial_$iface=`PRIV_NDD $NDD -get /dev/$iface instance 2>/dev/null`
done
```

is replaced with:

```bash
vars_fjqe="link_mode link_speed adv_autoneg_cap"
vars_fjgi="link_mode link_speed adv_autoneg_cap"
vars_igb="link_duplex link_speed adv_autoneg_cap"
LISTOFTYPES=`echo "$LINKNAMES" | sed 's/[0-9]*$//g' | sort -u`
LISTOFSETTYPES=`echo $LISTOFTYPES | sed -e 's/bge//' -e 's/dmfe//' -e s'/vsw//'`
for iface in $LISTOFSETTYPES; do
eval initial_$iface=`PRIV_NDD $NDD -get /dev/$iface instance 2>/dev/null`
done
```

The following code:

```bash
bge | dmfe ) #interfaces that do not need -set:
  for var in $vars
do
    echo 'NDD :' $NAME '::' $var '::' `PRIV_NDD $NDD -get /dev/$NAME $var
done
```

is replaced with:
getDirectoryListing

The following code:

```bash
P=%(path)s
PRIV_TEST -d "${P}" > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd "${P}"; PRIV_LS -al)
else
    echo 'PERMISSION DENIED'
fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
P=%(path)s
PRIV_TEST -d "${P}" > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x "${P}" -a -r "${P}" > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd "${P}" 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -al
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```

getDeviceInfo

The following code:

```bash
if [ -r /etc/release ]; then
    echo 'os:' `head -1 /etc/release 2>/dev/null`
else
```
is replaced with:

```
if [ -r /etc/release ]; then
  if grep Update /etc/release; then
    echo 'os:' `head -1 /etc/release 2>/dev/null`,' `grep Update /etc/release 2>/dev/null`
  else
    echo 'os:' `head -1 /etc/release 2>/dev/null`
  fi
else
  echo 'os:' `uname -sr 2>/dev/null`
fi
```

AIX

**getHostInfo**

The following code:

```
if [ `uname -v` -ge 6 ]; then
  # AIX 6 WPAR support
  wparid='uname -W 2>/dev/null'
  echo "wparid: $wparid"
  if [ $wparid -eq 0 -a -x /usr/sbin/lswpar ]; then
    echo 'begin lswpar:'
    /usr/sbin/lswpar -cqa name,state,type,hostname,directory,key,owner,application 2>/dev/null
    echo 'end lswpar'
  fi
fi
```

is replaced with:

```
if [ `uname -v` -ge 6 ]; then
  # AIX 6 WPAR support
  wparid='uname -W 2>/dev/null'
  echo "wparid: $wparid"
  if [ $wparid -eq 0 ]; then
    echo 'begin lswpar:'
    PRIV_LSWPAR /usr/sbin/lswpar -cqa name,state,type,hostname,directory,key,owner,application 2>/dev/null
    echo 'end lswpar'
  fi
fi
```
The following code:

```bash
# Don't output serial number if it isn't available. Also work around
# bug in prtconf on AIX 5.3 where there's no newline after the serial
# number output if it isn't available.
if [ "$serial" != " Not Available" -a "$serial" != "$serial" != " Not AvailableProcessor Type"
  echo 'serial:' $serial
fi
```

is replaced with:

```bash
# Don't output serial number if it isn't available. Also work around
# bug in prtconf on AIX 5.3 where there's no newline after the serial
# number output if it isn't available.
if [ "$serial" != "" -a "$serial" != " Not Available" -a "$serial" != " Not AvailableProcessor Type"
  echo 'candidate_serial[]:' $serial
else
  lsattr -E -l sys0 -a systemid 2>/dev/null | awk '{print "candidate_serial[]: " substr($2,7);}'
fi
```

The following code:

```
else
  lsattr -E -l sys0 -a modelname 2>/dev/null | awk '{print "model: " $2;}'
  lsattr -E -l sys0 -a systemid 2>/dev/null | awk '{print "serial: " $2;}'
  # Use this processor value if lparstat failed
```

is replaced with:

```
else
  lsattr -E -l sys0 -a modelname 2>/dev/null | awk '{print "model: " $2;}'
  lsattr -E -l sys0 -a systemid 2>/dev/null | awk '{print "candidate_serial[]: " substr(}'
  # Use this processor value if lparstat failed
```

The following code is added:

```
# Get candidate serial using lscfg -vp in case the above commands do not return any serial
lscfg -vp 2>/dev/null | grep -p "System VPD:" | grep "Cabinet Serial No" | sed -e 's/.'
**getIPAddresses**

The following code:

```bash
if `[ `uname -v` -ge 6 ]; then
    wparaddrs=`lswpar -Nqa address 2>/dev/null | xargs echo | sed -e 's/\./\./g'
fi
```

is replaced with:

```bash
if `[ `uname -v` -ge 6 ]; then
    wparaddrs=`PRIV_LSWPAR /usr/sbin/lswpar -Nqa address 2>/dev/null | xargs echo | sed -e 's/\./\./g' -e 's/ /|/g'
fi
```

**getDirectoryListing**

The following code:

```bash
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %(path)s; PRIV_LS -al)
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %(path)s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -al
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```
initialise

The following code is added:

```bash
# lswpar requires superuser privileges to get wpar information.
PRIV_LSWPAR() {
  "$@
}
```

Mac OS X

getAddressListing

The following code:

```bash
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  (cd %(path)s; PRIV_LS -alT)
else
  echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
  if [ $? -eq 0 ]; then
    (cd %(path)s 2> /dev/null
      if [ $? -eq 0 ]; then
        PRIV_LS -alT
      else
        echo 'PERMISSION DENIED'
      fi)
    else
      echo 'PERMISSION DENIED'
    fi
  else
    echo 'DIRECTORY NOT FOUND'
  fi
```
IRIX

getDirectoryListing

The following code:

```
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %(path)s; PRIV_LS -al)
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %(path)s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -al
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```

POWER HMC

getProcessList

The following code:

```
echo 'kernel:' `uname -r`
echo bios_level: `lshmc -b | grep bios= | cut -f2 -d=`
monhmc -r mem -n 0 | cut -f1 -d, | sed -e 's/Mem/ram/' -e 's/total//'`
```
echo 'kernel:' `uname -r`
echo 'model:' `lshmc -v | grep '^*TM' | cut -f2 -d' ''
echo 'serial:' `lshmc -v | grep '^*SE' | cut -f2 -d' ''
echo 'bios_level:' `lshmc -b | grep bios= | cut -f2 -d' ''
monhmc -r mem -n 0 | cut -f1 -d, | sed -e 's/Mem/logical_ram/' -e 's/total//' getHostInfo

The following code:

echo 'kernel:' `uname -r`
echo 'model:' `lshmc -v | grep '^*TM' | cut -f2 -d' ''
echo 'serial:' `lshmc -v | grep '^*SE' | cut -f2 -d' ''
echo 'bios_level:' `lshmc -b | grep bios= | cut -f2 -d' ''
monhmc -r mem -n 0 | cut -f1 -d, | sed -e 's/Mem/logical_ram/' -e 's/total//' is replaced with:

echo 'kernel:' `uname -r`
echo 'model:' `lshmc -v | grep '^*TM' | cut -f2 -d' ''
echo 'serial:' `lshmc -v | grep '^*SE' | cut -f2 -d' ''
echo 'bios_level:' `lshmc -b | grep bios= | cut -f2 -d' ''
monhmc -r mem -n 0 | cut -f1 -d, | sed -e 's/Mem/logical_ram/' -e 's/total//' Tru64
getProcessList

The following code:

echo 'kernel:' `uname -v 2>/dev/null`
echo 'model:' `PRIV_HWMGR hwmgr get attr -cat platform -a name 2>/dev/null | grep name | s vmstat -P 2>/dev/null | grep "Total Physical Memory" | sed -e 's/Total Physical Memory *= */sys_serial_num *= */serial: /'
psrinfo -v
echo 'end tru64_psrinfo'
is replaced with:

echo 'kernel:' `uname -v 2>/dev/null`
echo 'model:' `PRIV_HWMGR hwmgr get attr -cat platform -a name 2>/dev/null | grep name | s echo 'ram:' `vmstat -P 2>/dev/null | grep "^Total Physical Memory" | cut -f2 -d'`
```
echo "serial: \`/sbin/consvar -g sys_serial_num 2>/dev/null | cut -f2- -d=`
echo 'begin tru64_psrinfo:'
psrinfo -v
echo 'end tru64_psrinfo'
```

**getDirectoryListing**

The following code:

```
PRIV_TEST -d %{path}s -a -r %{path}s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %{path}s; PRIV_LS -al)
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```
PRIV_TEST -d %{path}s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %{path}s -a -r %{path}s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %{path}s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -al
        else
            echo 'PERMISSION DENIED'
        fi
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'PERMISSION DENIED'
fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```

**OpenVMS**

**getHostInfo**

The following code:

```
WRITE SYS$OUTPUT "arch: ", F$GETSYI("ARCH_NAME")
WRITE SYS$OUTPUT "model: ", F$GETSYI("HW_NAME")
WRITE SYS$OUTPUT "ram: ", F$GETSYI("MEMSIZE")/(1048576/F$GETSYI("PAGE_SIZE")), "MB"
```
WRITE SYS$OUTPUT "begin show-cpu:"
PIPE SHOW CPU /FULL | SEARCH SYS$INPUT /NOHIGHLIGHT "COThd:","Type","Speed"
WRITE SYS$OUTPUT "end show-cpu:"

is replaced with:

WRITE SYS$OUTPUT "arch: ", F$GETSYI("ARCH_NAME")
WRITE SYS$OUTPUT "model: ", F$GETSYI("HW_NAME")
WRITE SYS$OUTPUT "logical_ram: ", F$GETSYI("MEMSIZE")/(1048576/F$GETSYI("PAGE_SIZE")), "MB"
WRITE SYS$OUTPUT "begin show-cpu:"
PIPE SHOW CPU /FULL | SEARCH SYS$INPUT /NOHIGHLIGHT "COThd:","Type","Speed"
WRITE SYS$OUTPUT "end show-cpu:"

VMware ESXi

getAddressListing

The following code:

PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  (cd %(path)s; PRIV_LS -lae --color=never)
else
  echo 'DIRECTORY NOT FOUND'
fi

is replaced with:

PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
  if [ $? -eq 0 ]; then
    (cd %(path)s 2> /dev/null
    if [ $? -eq 0 ]; then
      PRIV_LS -lae --color=never
    else
      echo 'PERMISSION DENIED'
    fi)
  else
    echo 'PERMISSION DENIED'
  fi
else
  echo 'DIRECTORY NOT FOUND'
fi
UnixWare

getDirectoryListing

The following code:

```
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  (cd %(path)s; PRIV_LS -al)
else
  echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
  if [ $? -eq 0 ]; then
    (cd %(path)s 2> /dev/null
      if [ $? -eq 0 ]; then
        PRIV_LS -al
      else
        echo 'PERMISSION DENIED'
      fi)
  else
    echo 'PERMISSION DENIED'
  fi
else
  echo 'DIRECTORY NOT FOUND'
fi
```

FreeBSD

getHostInfo

The following code is added:

```
if [ -x /sbin/sysctl ]; then
  ram=`/sbin/sysctl hw.realmem 2>/dev/null | sed -n 's/hw.realmem *[=-] */p'`
  if [ "$ram" != "" ]; then
    echo "ram: $ram"
  fi
  logical_ram=`/sbin/sysctl hw.physmem 2>/dev/null | sed -n 's/hw.physmem *[=-] */p'`
  if [ "$logical_ram" != "" ]; then
    echo "logical_ram: $logical_ram"
```
**getDirectoryListing**

The following code:

```bash
PRIV_TEST -d %s -a -r %s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %s; PRIV_LS -alT)
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
PRIV_TEST -d %s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %s -a -r %s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -alT
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'PERMISSION DENIED'
else
    echo 'DIRECTORY NOT FOUND'
fi
```

**Linux**

**getHostInfo**

The following code:

```bash
ram=`awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null`
```

is replaced with:

```bash
ram='awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null'
```
The following code:

```bash
logical_ram=`awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null`
```

is replaced with:

```bash
logical_ram=`awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null`
```

The following code:

```bash
ram=`grep "Physical Mem\." ${file} | sed 's/[0-9]*//g'`B
uuid=`grep "BIOS UUID\." ${file}`
else
    ram=`grep "Physical Mem\." ${file} | sed 's/[0-9]*//g'`B
    # For esx/esxi, we should NOT use memory from dmesg or /proc/meminfo
    # because the values are incorrect
    logical_ram=""
    uuid=`grep "BIOS UUID\." ${file}`
else
```

is replaced with:

```bash
ram=`grep "Physical Mem\." ${file} | sed 's/[0-9]*//g'`B
# For esx/esxi, we should NOT use memory from dmesg or /proc/meminfo
# because the values are incorrect
logical_ram=""
uuid=`grep "BIOS UUID\." ${file}`
else
```

The following code:

```bash
uuid=`PRIV_XE $XE host-list | grep uuid | head -n 1 | cut -f2 -d: | awk '{print $1}'`
ram=`PRIV_XE $XE host-param-get uuid=$uuid param-name=memory-total`
print=1
fi
fi
```

is replaced with:

```bash
uuid=`PRIV_XE $XE host-list | grep uuid | head -n 1 | cut -f2 -d: | awk '{print $1}'`
logical_ram=`PRIV_XE $XE host-param-get uuid=$uuid param-name=memory-total`
print=1
fi
fi
```

The following code:

```bash
if [ "$ram" != "" ]; then
    echo 'ram:' $ram
fi
```

```bash
fi
```

```bash
fi
```
is replaced with:

```bash
if [ "${ram}" != "" ]; then
echo 'ram:' ${ram}
fi
if [ "${logical_ram}" != "" ]; then
echo 'logical_ram:' ${logical_ram}
fi
```

The following code:

```bash
# candidate from the values returned, where "best" is the first non-bogus value
if [ -x /usr/bin/lshal ]; then
    /usr/bin/lshal 2>/dev/null | sed -e 's/(string)$//g' -e "s/"//g" | awk '    
    $1 ~ /system.hardware.serial/ {
        sub(".*system.hardware.serial = */, "", "");
        printf("candidate_serial[]: %s\n", $0);
    }
    $1 ~ /system.hardware.uuid/ {
        sub(".*system.hardware.uuid = */, "", "");
        printf("candidate_uuid[]: %s\n", $0);
    }
    $1 ~ /system.hardware.product/ {
        sub(".*system.hardware.product = *", ",");
        printf("candidate_model[]: %s\n", $0);
    }
    $1 ~ /system.hardware.vendor/ {
        sub(".*system.hardware.vendor = *", ",");
        printf("candidate_vendor[]: %s\n", $0);
    }
' 
fi
```

is replaced with:

```bash
# candidate from the values returned, where "best" is the first non-bogus value
if [ -x /usr/bin/lshal ]; then
    /usr/bin/lshal 2>/dev/null | sed -e 's/(string)$//g' -e "s/"//g" | awk '    
    $1 ~ /(smbios\.system|system\.hardware).serial/ {
        sub(/.*(smbios\.system|system\.hardware).serial = */, "", "");
        printf("candidate_serial[]: %s\n", $0);
    }
    $1 ~ /(smbios\.system|system\.hardware).uuid/ {
        sub(/.*(smbios\.system|system\.hardware).uuid = */, "", "");
```

```bash
```
printf("candidate_uuid[]: %s\n", $0);
}
$1 ~ /(smbios\system|system\hardware)\product/ {
    sub(/.*(smbios\system|system\hardware)\product = */, "");
    printf("candidate_model[]: %s\n", $0);
}
$1 ~ /system\(\hardware)?\vendor/ {
    sub(/.*(system|system\hardware)\vendor = */, "");
    printf("candidate_vendor[]: %s\n", $0);
}'
fi

getDirectoryListing
The following code:

PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %(path)s; PRIV_LS -a --full-time --color=never)
else
    echo 'DIRECTORY NOT FOUND'
fi

is replaced with:

PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %(path)s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -a --full-time --color=never
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'PERMISSION DENIED'
else
    echo 'DIRECTORY NOT FOUND'
fi

OpenBSD

getHostInfo
The following code is added:
if [ -x /sbin/sysctl ]; then
  logical_ram="/sbin/sysctl hw.physmem 2>/dev/null | sed -n 's/hw.physmem *=*/p'
  if [ "$logical_ram" != "" ]; then
    echo "logical_ram: $logical_ram"
  fi
fi

getDirectoryListing

The following code:

PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  (cd %(path)s; PRIV_LS -aLT)
else
  echo 'DIRECTORY NOT FOUND'
fi

is replaced with:

PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
  if [ $? -eq 0 ]; then
    (cd %(path)s 2> /dev/null
      if [ $? -eq 0 ]; then
        PRIV_LS -aLT
      else
        echo 'PERMISSION DENIED'
      fi)
    else
      echo 'PERMISSION DENIED'
    fi
  else
    echo 'PERMISSION DENIED'
  fi
else
  echo 'DIRECTORY NOT FOUND'
fi

NetBSD

getHostInfo

The following code is added:
if [ -x /sbin/sysctl ]; then
    logical_ram=`/sbin/sysctl hw.physmem64 2>/dev/null | sed -n 's/hw.physmem64 *[\:=] */p`
    if [ "$logical_ram" != "" ]; then
        echo "logical_ram: $logical_ram"
    else
        logical_ram=`/sbin/sysctl hw.physmem 2>/dev/null | sed -n 's/hw.physmem *[\:=] */p`
        if [ "$logical_ram" != "" ]; then
            echo "logical_ram: $logical_ram"
        fi
    fi
fi

getDirectoryListing

The following code:

PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %(path)s; PRIV_LS -alT)
else
    echo 'DIRECTORY NOT FOUND'
fi

is replaced with:

PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %(path)s 2> /dev/null
            if [ $? -eq 0 ]; then
                PRIV_LS -alT
            else
                echo 'PERMISSION DENIED'
            fi)
        else
            echo 'PERMISSION DENIED'
        fi
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'DIRECTORY NOT FOUND'
fi
HP-UX

getMACAddresses

The following code:

```bash
echo 'end machinfo:'
else
  cores=0
  ram=0
  if [ -x /usr/sbin/parstatus ]; then
    par_id=/usr/sbin/parstatus -w 2> /dev/null | grep "The local partition number is"
    if [ "$par_id" ]; then
      echo par_id: $par_id
      cpumem=/usr/sbin/parstatus -C -M | awk '{PAR_ID=$par_id} BEGIN{ FS=":" }{ if (NF==2) { cpu+=cpuarray[1]; mem+=int(memarray[1])} };END{ printf "%d %d
",cpu,mem }'
      cores=`echo "$cpumem" |awk '{print $1}'`
      logical_ram=`echo "$cpumem" |awk '{print $2}'`
      logical_ram=`echo "$logical_ram" * 1048576 * 1024 | bc`
      echo logical_ram: $logical_ram
    fi
  fi
fi
```

is replaced with:

```bash
echo 'end machinfo:'
else
  cores=0
  logical_ram=""
  ram=""
  if [ -x /usr/sbin/parstatus ]; then
    par_id=/usr/sbin/parstatus -w 2> /dev/null | grep "The local partition number is"
    if [ "$par_id" ]; then
      echo par_id: $par_id
      cpumem=/usr/sbin/parstatus -C -M | awk '{PAR_ID=$par_id} BEGIN{ FS=":" }{ if (NF==2) { cpu+=cpuarray[1]; mem+=int(memarray[1])} };END{ printf "%d %d
",cpu,mem }'
      cores=`echo "$cpumem" |awk '{print $1}'`
      logical_ram=`echo "$cpumem" |awk '{print $2}'`
      logical_ram=`echo "$logical_ram" * 1048576 * 1024 | bc`
      echo logical_ram: $logical_ram
    fi
  fi
fi
```

The following code:

```bash
if [ $ram -eq 0 ]; then
  ram=`echo "selclass qualifier memory;infolog" | cstm 2>/dev/null | grep 'System To`
if [ "$ram" != "" ]; then
```
echo $ram | sed -e 's/.*\(([^\-]+)\)[^0-9]*\([0-9]*\).*$/ram: \2\1/'
fi
if [ "$ram" = "" ]; then # cstm not installed or do not have suitable credentials,
  # first try to check the Ignite-UX manifest
  MANIFEST_PATH="/var/opt/ignite/local/manifest/manifest"
  if [ -r $MANIFEST_PATH ]; then
    ram=`grep "Main Memory" "$MANIFEST_PATH" | cut -d: -f2 | sed 's/ //g'`
    if [ "$ram" != "" ]; then
      echo "ram: $ram"
    fi
  fi
  fi
  fi
if [ "$ram" = "" ]; then # cannot read manifest or cannot extract RAM info, try to read syslog
  SYSLOG_PATH="/var/adm/syslog/syslog.log"
  if [ -r $SYSLOG_PATH ]; then
    ram=`grep "Physical" "$SYSLOG_PATH" | sed 's/^.*Physical: *\([0-9]*\).*\([GMK]\)bytes.*$/\1\2B/'`
    if [ "$ram" != "" ]; then
      echo "ram: $ram"
    fi
  fi
  fi
  fi
is replaced with:

ram=`echo "selclass qualifier memory;infolog" | cstm 2>/dev/null | grep 'System Total'`
if [ "$ram" != "" ]; then
  echo $ram | sed -e 's/.*\(([^\-]+)\)[^0-9]*\([0-9]*\).*$/ram: \2\1/'
fi
if [ "$logical_ram" = "" ]; then
  # Get logical_ram from manifest for systems other than vpar
  if [ "$spar_id" = "" ]; then
    # first try to check the Ignite-UX manifest
    MANIFEST_PATH="/var/opt/ignite/local/manifest/manifest"
    if [ -r $MANIFEST_PATH ]; then
      logical_ram=`grep "Main Memory" "$MANIFEST_PATH" | cut -d: -f2 | sed 's/ //g'`
      if [ "$logical_ram" != "" ]; then
        echo "logical_ram: $logical_ram"
      fi
    fi
  fi
  fi
  fi
if [ "$logical_ram" = "" ]; then # cannot read manifest or cannot extract RAM info, try to read syslog
  SYSLOG_PATH="/var/adm/syslog/syslog.log"
  if [ -r $SYSLOG_PATH ]; then
    logical_ram=`grep "Physical" "$SYSLOG_PATH" | sed 's/^.*Physical: *\([0-9]*\).*\([GMK]\)bytes.*$/\1\2B/'`
    if [ "$logical_ram" != "" ]; then
      echo "logical_ram: $logical_ram"
    fi
  fi
  fi
  fi
getDirectoryListing

The following code:

```bash
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  (cd %(path)s; PRIV_LS -al)
else
  echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
  PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
  if [ $? -eq 0 ]; then
    (cd %(path)s 2> /dev/null
      if [ $? -eq 0 ]; then
        PRIV_LS -al
      else
        echo 'PERMISSION DENIED'
      fi)
    else
      echo 'PERMISSION DENIED'
    fi
  else
    echo 'PERMISSION DENIED'
  fi
else
  echo 'DIRECTORY NOT FOUND'
fi
```

VMware ESX

getHostInfo

The following code:

```bash
ram=`awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null`
```

is replaced with:

```bash
logical_ram=`awk '/^MemTotal:/ {print $2 "KB"}' /proc/meminfo 2>/dev/null`
```
The following code is added:

```bash
# For esx/esxi, we should NOT use memory from dmesg or /proc/meminfo
# because the values are incorrect
logical_ram=""
```

The following code:

```bash
# /proc/meminfo reports incorrectly for Xen domains, use "xe"
uuid=`PRIV_XE $XE host-list | grep uuid | head -n 1 | cut -f2 -d: | awk '{print $1}'
ram=`PRIV_XE $XE host-param-get uuid=$uuid param-name=memory-total`
print=1
fi
fi
```

is replaced with:

```bash
# /proc/meminfo reports incorrectly for Xen domains, use "xe"
uuid=`PRIV_XE $XE host-list | grep uuid | head -n 1 | cut -f2 -d: | awk '{print $1}'
logical_ram=`PRIV_XE $XE host-param-get uuid=$uuid param-name=memory-total`
print=1
fi
fi
```

The following code:

```bash
fi
if [ "${ram}" != "" ]; then
    echo 'ram:' ${ram}
fi
fi
```

is replaced with:

```bash
fi
if [ "${ram}" != "" ]; then
    echo 'ram:' ${ram}
fi
if [ "${logical_ram}" != "" ]; then
```
echo 'logical_ram: '${logical_ram}
fi
fi

The following code:

```bash
# candidate from the values returned, where "best" is the first non-bogus value
if [-x /usr/bin/lshal ]; then
    /usr/bin/lshal 2>/dev/null | sed -e 's/(string)$//g' -e "s/'//g" | awk '
$1 ~ /system.hardware.serial/ {
    sub(".*system.hardware.serial = *", "");
    printf("candidate_serial[]: %s\n", $0);
}
$1 ~ /system.hardware.uuid/ {
    sub(".*system.hardware.uuid = *", "");
    printf("candidate_uuid[]: %s\n", $0);
}
$1 ~ /system.hardware.product/ {
    sub(".*system.hardware.product = *", "");
    printf("candidate_model[]: %s\n", $0);
}
$1 ~ /system\.hardware\.vendor/ {
    sub(".*system\.hardware\.vendor = *", "");
    printf("candidate_vendor[]: %s\n", $0);
}'
fi
```

is replaced with:

```bash
# candidate from the values returned, where "best" is the first non-bogus value
if [-x /usr/bin/lshal ]; then
    /usr/bin/lshal 2>/dev/null | sed -e 's/(string)$//g' -e "s/'//g" | awk '
$1 ~ /smbios\.system\.serial/ {
    sub(".*(smbios\.system|system\.hardware).serial = */, "");
    printf("candidate_serial[]: %s\n", $0);
}
$1 ~ /smbios\.system\.uuid/ {
    sub(".*(smbios\.system|system\.hardware).uuid = */, "");
    printf("candidate_uuid[]: %s\n", $0);
}
$1 ~ /smbios\.system\.product/ {
    sub(".*(smbios\.system|system\.hardware).product = */, "");
    printf("candidate_model[]: %s\n", $0);
}
$1 ~ /system\.hardware\.vendor/ {
    sub(".*(system|system\.hardware).vendor = */, "");
    printf("candidate_vendor[]: %s\n", $0);
}'
fi
```
getDirectoryListing

The following code:

```bash
PRIV_TEST -d %(path)s -a -r %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    (cd %(path)s; PRIV_LS -a --full-time --color=never)
else
    echo 'DIRECTORY NOT FOUND'
fi
```

is replaced with:

```bash
PRIV_TEST -d %(path)s > /dev/null 2> /dev/null
if [ $? -eq 0 ]; then
    PRIV_TEST -x %(path)s -a -r %(path)s > /dev/null 2> /dev/null
    if [ $? -eq 0 ]; then
        (cd %(path)s 2> /dev/null
        if [ $? -eq 0 ]; then
            PRIV_LS -a --full-time --color=never
        else
            echo 'PERMISSION DENIED'
        fi)
    else
        echo 'PERMISSION DENIED'
    fi
else
    echo 'DIRECTORY NOT FOUND'
fi
```

Discovery command changes from 9.0 SP1 to 9.0 SP2

AIX

getHostInfo

The following code:

```bash
# Extract CPU info from lparstat
num_physical_processors=`egrep '^Active Physical CPUs in system' /tmp/tideway.$$ | cut -f2`
num_logical_processors=`egrep '^Online Virtual CPUs' /tmp/tideway.$$ | cut -f2 -d:`
is replaced with:

```bash
# Extract CPU info from lparstat
num_logical_processors=`egrep '^Online Virtual CPUs' /tmp/tideway.$$ | cut -f2 -d:`
```

The following code:

```bash
    echo "cpu_threading_enabled:" $smt_enabled
fi
if [ "$num_physical_processors" != "" ]; then
    echo "num_processors": $num_physical_processors
fi
if [ "$num_logical_processors" != "" ]; then
    echo "num_logical_processors:" $num_logical_processors
fi
```

is replaced with:

```bash
    echo "cpu_threading_enabled:" $smt_enabled
fi
if [ "$num_logical_processors" != "" ]; then
    echo "num_logical_processors:" $num_logical_processors
fi
```

**Linux**

**getHostInfo**

The following code is added:

```bash
if [ -d /sys/class/dmi/id ]; then
    echo "candidate_model[]": "`cat /sys/class/dmi/id/product_name 2>/dev/null`"
    echo "candidate_serial[]": "`PRIV_CAT /sys/class/dmi/id/product_serial 2>/dev/null`"
    echo "candidate_uuid[]": "`PRIV_CAT /sys/class/dmi/id/product_uuid 2>/dev/null`"
    echo "candidate_vendor[]": "`cat /sys/class/dmi/id/sys_vendor 2>/dev/null`"
fi
```
getNetworkInterfaces

The following code:

```
ifconfig -a 2>/dev/null
ETHTOOL=""
if [ -f /sbin/ethtool ]; then
```

is replaced with:

```
ifconfig -a 2>/dev/null
if [ $? -eq 0 ]; then
ETHTOOL=""
if [ -f /sbin/ethtool ]; then
```

The following code:

```
echo 'begin details:'
for i in `ifconfig -a 2>/dev/null | egrep '^[a-z]' | awk '{print $1;}'`
do
```

is replaced with:

```
echo 'begin details:'
for i in `ifconfig -a 2>/dev/null | egrep '^[a-z]' | awk -F: '{print $1;}'`
do
```

The following code:

```
done
echo 'end details:'
```

is replaced with:

```
```
```
done
echo 'end details:'
fi

HP-UX

getMACAddresses

The following code:

```bash
lanscan -am
```

is replaced with:

```bash
lanscan | grep ETHER | awk '{print $2, $8;}'
```

VMware ESX

getHostInfo

The following code is added:

```bash
if [ -d /sys/class/dmi/id ]; then
    echo "candidate_model[]: " `cat /sys/class/dmi/id/product_name 2>/dev/null`
    echo "candidate_serial[]: " `PRIV_CAT /sys/class/dmi/id/product_serial 2>/dev/null`
    echo "candidate_uuid[]: " `PRIV_CAT /sys/class/dmi/id/product_uuid 2>/dev/null`
    echo "candidate_vendor[]: " `cat /sys/class/dmi/id/sys_vendor 2>/dev/null`
fi
```

getNetworkInterfaces

The following code:

```bash
ifconfig -a 2>/dev/null
ETHTOOL=""
```

is replaced with:

```bash
```
The following code:

```bash
ifconfig -a 2>/dev/null
if [ $? -eq 0 ]; then
    ETHTOOL=""

The following code:

```bash
    echo 'begin details:'
    for i in `ifconfig -a 2>/dev/null | egrep '^\[a-z\]' | awk '{print $1;}'`
    do
        ...
    done
```

is replaced with:

```bash
    echo 'begin details:'
    for i in `ifconfig -a 2>/dev/null | egrep '^\[a-z\]' | awk -F: '{print $1;}'`
    do
        ...
    done
```

Discovery command changes from 9.0 and 9.0 SP1

Solaris

getHostInfo

The following code is added:

```bash
if [ -x /usr/sbin/virtinfo ]; then
    # LDOM support for Solaris 11 in system/core-os, and Solaris 9/10 in SUNWcsu
    echo 'begin virtinfo:'
    /usr/sbin/virtinfo -ap 2>/dev/null
    echo 'end virtinfo:'
fi
```

AIX

getHBAList

The following code:

```bash
do
    adapter_name=`echo $i | cut -f1 -d:`
    adapter_status=`echo $i | cut -f2 -d:`
    adapter_type=`echo $i | cut -f3 -d:`
```
if [ $adapter_status = "Available" ]; then
  echo begin lscfg-vl-$adapter_name:
  lscfg -vl $adapter_name

is replaced with:

```
do
  adapter_name=`echo $i | cut -f1 -d:`
  adapter_status=`echo $i | cut -f2 -d:`
  adapter_type=`echo $i | cut -f3 -d: | sed -e 's/,/./'`
  if [ $adapter_status = "Available" ]; then
    echo begin lscfg-vl-$adapter_name:
    lscfg -vl $adapter_name
```

Mainframe

The following method is added:

- getTransactionProgram

HPUX

The following method is added

- getHBAList

initialise

The following code is added:

```
# fcmsutil requires superuser privileges to display any HBA information
PRIV_FCMSUTIL() {
  "$@
}
```

Discovery command changes from 8.3 SP2 to 9.0

Solaris

The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
getNetworkInterfaces
getIPAddresses

getProcessList

The following code:

```bash
if [ `uname -r | cut -f2 -d.` -ge 10 ]  && [ -x /usr/bin/zonename ]; then
    zone=`/usr/bin/zonename 2>/dev/null`
    ps -eo pid,ppid,uid,user,zone,args 2>/dev/null | awk "\$5~/^($zone|ZONE)$/ {print}"
else
    ps -eo pid,ppid,uid,user,args 2>/dev/null
fi
```

is replaced with:

```bash
os_ver=`uname -r | cut -d. -f2`
if [ $os_ver -ge 10 -a -x /usr/bin/zonename ]; then
    zone=`/usr/bin/zonename 2>/dev/null`
    ps -eo pid,ppid,uid,user,zone,args 2>/dev/null | awk "\$5~/^($zone|ZONE)$/ {print}"
else
    ps -eo pid,ppid,uid,user,args 2>/dev/null
fi
```

The following code:

```bash
PRIV_PS /usr/ucb/ps -axww 2>/dev/null
```

is replaced with:

```bash
if [ $os_ver -ge 11 ]; then
    PRIV_PS /usr/bin/ps axww 2>/dev/null
else
    if [ -x /usr/ucb/ps ]; then
        PRIV_PS /usr/ucb/ps -axww 2>/dev/null
    fi
fi
```

```bash
if [ -x /usr/bin/pargs -a -d /proc ]; then
    echo begin pargs:
    PRIV_PARGS /usr/bin/pargs `ls -1 /proc` 2>/dev/null
    echo end pargs:
fi
```
getNetworkConnectionList

The following code is added:

```bash
netstat -anf inet6 2>/dev/null | grep -v '^ \*.*\.*$
```

getHostInfo

The following code:

```bash
showrev 2>/dev/null | nawk -F: '/^Kernel version/ {gsub("^ *", "", $2); print "kernel:", $2; exit}' 

`uname -i 2>/dev/null`

/usr/sbin/prtconf 2>/dev/null | nawk '/^Memory size:/ {print "ram: " $3 "MB"}''
```

is replaced with:

```bash
os_ver=`uname -r | cut -d. -f2`
if [ $os_ver -ge 11 ]; then
    pkg list -H --no-refresh system/kernel | awk '{print "kernel:", $2; exit}'
else
    showrev 2>/dev/null | nawk -F: '/^Kernel version/ {gsub("^ *", "", $2); print "kernel:", $2; exit}'
fi

`uname -i 2>/dev/null`

/usr/sbin/prtconf 2>/dev/null | nawk '/^Memory size:/ {print "ram: " $3 "MB"}''
```

The following code:

```bash
fi

echo 'hostid: ' `hostid`
if [ -f /etc/ssphostname ]; then
    # E10K support
    echo 'ssphostname: ' `cat /etc/ssphostname`
fi
```

is replaced with:

```bash
fi
```
echo 'hostid:' `hostid`
if [ -r /etc/ssphostname ]; then
  # E10K support
  echo 'ssphostname:' `cat /etc/ssphostname`
fi

getPackageList
The following code:

pkginfo -l 2>/dev/null

is replaced with:

os_ver=`uname -r | cut -d. -f2`
if [ $os_ver -ge 11 ]; then
  echo begin pkg_list:
  echo arch: `uname -p`
  pkg list -H --no-refresh 2>/dev/null
  echo end pkg_list:
fi
PKGINFO=`tw_which pkginfo`
if [ ! -z "$PKGINFO" -a -x $PKGINFO ]; then
  pkginfo -l 2>/dev/null
fi

initialise
The following code:

ls "$@"
}

is replaced with:

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
  "$@
}

tw_which() {
  SAVE=$IFS

BMC Atrium Discovery 10.2
The following code:

```bash
IFS=:
    "$@

PRIV_NDD() {
    "$@
}

# /usr/ucb/ps requires superuser privileges to display full command line
# information (without this, command lines will be limited to 80 characters).
# This affects Solaris 10 and Solaris 8 & 9 with certain patches
```

is replaced with:

```bash
# dladm requires superuser privileges to display speed, duplex settings, and
# port aggregation information
PRIV_DLADM() {
    "$@
}

# ndd requires superuser privileges to display any interface speed
# and negotiation settings
PRIV_NDD() {
    "$@
}

# By default, the standard ps command on Solaris will truncate command lines
# to 80 characters. This affects Solaris 11, Solaris 10 and Solaris 8 & 9
# with certain patches.
#
# In order to display unlimited command lines, there are several options:
#
# pargs - This tool is available in more recent updates of Solaris 9 and
# all updates of Solaris 10 and later. This tool requires the
# proc_owner privilege to display unlimited command lines for all
# processes.
#
# /usr/bin/ps - On Solaris 11, the standard ps command can display
# unlimited command lines by using BSD style command line arguments.
# This still requires the the tool is run with proc_owner privilege
#
# /usr/ucb/ps - This tool is part of the UCB compatibility package which is
# usually installed by default on versions up to and including
# Solaris 10 (SUNWscpu package). This tool requires the
# proc_owner privilege to display unlimited command lines for all
# processes.
# On Solaris 11, the compatibility/ucb is not usually installed
# by default and in any case, the /usr/ucb/ps command is simply
# a link to /usr/bin/ps
#
# In order for the Discovery Condition pattern to correctly detect whether
# ps is being executed with proc_owner privilege, this function must accept
# both the ps command and the ppriv command used by pattern.
 PRIV_PS() {
  "$@
}

# See comments above PRIV_PS, above
 PRIV_PARGS() {
  "$@
}

getFileSystems

The following code:

echo begin df:
  df -lk 2>/dev/null
echo end df:
echo begin mount:
  if [ -f /etc/mnttab ]; then
    cat /etc/mnttab
  fi
echo end mount:
echo begin xtab:
  if [ -f /etc/xtab ]; then
    cat /etc/xtab
  fi
echo end xtab:
oformat}
s replaced with:
oformat}
echo begin df:
  PRIV_DF df -lk 2>/dev/null
echo end df:
echo begin mount:
  if [ -r /etc/mnttab ]; then
    cat /etc/mnttab
  fi
echo end mount:
echo begin xtab:
  if [ -r /etc/xtab ]; then
    cat /etc/xtab
  fi
The following code:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
    if [ -f $configfile ]; then
        cat $configfile
    fi
fi
```

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
    if [ -r $configfile ]; then
        cat $configfile
    fi
fi
```

getDeviceInfo
The following code:

```bash
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(search|domain)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`
if [ -f /etc/release ]; then
    echo 'os:' `head -1 /etc/release 2>/dev/null`
else
    echo 'os:' `uname -sr 2>/dev/null`
```

is replaced with:

```bash
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(search|domain)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`
if [ -f /etc/release ]; then
    echo 'os:' `head -1 /etc/release 2>/dev/null`
else
    echo 'os:' `uname -sr 2>/dev/null`
```
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [-r /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(search|domain)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`
if [-r /etc/release ]; then
    echo 'os:' `head -1 /etc/release 2>/dev/null`
else
    echo 'os:' `uname -sr 2>/dev/null`
fi

getchannel
The following code:

showrev -p 2>/dev/null | grep -v "No patches are installed" | cut -c-16 | nawk '{print $2;}

getPatchList
is replaced with:

os_ver=`uname -r | cut -d. -f2`
if [ $os_ver -lt 11 ]; then
    showrev -p 2>/dev/null | grep -v "No patches are installed" | cut -c-16 | nawk '{print $2;}
else
    echo NO PATCHES
fi

AIX
The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses

The host_info command is removed from the getPatchList method.
The lsof command is removed from the getPatchList method.

 initialise
The following code:
ls "@$"
)

# lslpp requires superuser privileges to list all installed packages
PRIV_LSLPP() {
  "@$"
}

is replaced with:

ls "@$"
)

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
  "@$"
}

# lslpp requires superuser privileges to list all installed packages
PRIV_LSLPP() {
  "@$"
}

getNetworkConnectionList

The following code:

if [ `uname -v` -ge 6 ]; then
  if [ `uname -W` -eq 0 ]; then
    netstat -an -f inet -o 2>/dev/null | egrep "Global|Proto" | sed -e 's/Global //'
  else
    netstat -an -f inet 2>/dev/null
  fi
else
  netstat -an -f inet 2>/dev/null
fi

is replaced with:

if [ `uname -v` -ge 6 ]; then
  if [ `uname -W` -eq 0 ]; then
    netstat -an -f inet -o 2>/dev/null | egrep "Global|Proto" | sed -e 's/Global //'
    netstat -an -f inet6 -o 2>/dev/null
  else
    netstat -an -f inet 2>/dev/null
  fi

getDeviceInfo

The following code:

```
ihn='hostname 2>/dev/null'
echo 'hostname:' $ihn
if [-f /etc/resolv.conf ]; then
   echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`
# VIO support
if [-f /usr/ios/cli/ios.level ]; then
   viover='cat /usr/ios/cli/ios.level'
   os="VIO $viover"
else
```

is replaced with:

```
ihn='hostname 2>/dev/null'
echo 'hostname:' $ihn
if [-r /etc/resolv.conf ]; then
   echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`
# VIO support
if [-r /usr/ios/cli/ios.level ]; then
   viover='cat /usr/ios/cli/ios.level'
   os="VIO $viover"
else
```

getHBAList

The following code:

```
if [ $adapter_status = "Available" ]; then
   echo begin lscfg-vl-$adapter_name:
   lscfg -vl $adapter_name
   echo end lscfg-vl-$adapter_name:
```
```
  echo begin lspp-$adapter_name:
  lsllp -L devices.*.$adapter_type.*
```

is replaced with:

```
if [ $adapter_status = "Available" ]; then
  echo begin lscfg-vl-$adapter_name:
  lscfg -vl $adapter_name
  echo $? > /dev/null # Prevent terminal issues
  echo end lscfg-vl-$adapter_name:
  echo begin lspp-$adapter_name:
  lsllp -L devices.*.$adapter_type.*

getFileSystems

The following code:

```
  echo begin df:
  if [ -x /usr/sysv/bin/df ]; then
    /usr/sysv/bin/df -lg 2>/dev/null
  else
    df -k 2>/dev/null
  fi
  echo end df:
  echo begin mount:
  mount 2>/dev/null
  echo end mount:
  echo begin xtab:
  if [ -f /etc/xtab ]; then
    cat /etc/xtab
  fi
  echo end xtab:
```

is replaced with:

```
echo begin df:
if [ -x /usr/sysv/bin/df ]; then
  PRIV_DF /usr/sysv/bin/df -lg 2>/dev/null
else
  PRIV_DF df -k 2>/dev/null
fi
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
echo end xtab:
```
The following code:

```
echo begin xtab:
if [ -r /etc/xtab ]; then
  cat /etc/xtab
fi
echo end xtab:
```

is replaced with:

```
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```

Windows

The following SSH discovery methods are removed:

- getProcessList
- getProcessToConnectionMapping
- getHostInfo
- getInterfaceList
- getDirectoryListing
- initialise
- getNetworkConnectionList
- getFileMetadata
- getDeviceInfo
- getPatchList
- getFileContent
Mac OS X

The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses

getNetworkConnectionList

The following code:

```
netstat -an -f inet 2>/dev/null
```

is replaced with:

```
netstat -an -f inet 2>/dev/null
netstat -an -f inet6 -l 2>/dev/null
```

initialise

The following code:

```
PRIV_LS() {
    ls "$@
}
```

is replaced with:

```
PRIV_LS() {
    ls "$@
}

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
    "$@
}
```
getDeviceInfo

The following code:

```bash
echo "hostname:" `hostname -s 2>/dev/null`
system_profiler SPHardwareDataType SPSoftwareDataType > /tmp/tideway-hw-$$. 2>/dev/null
grep "Computer Name: " /tmp/tideway-hw-$$. | sed 's/^.*: \(.*\)/description: \1/'
if [ -f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf`
fi
grep "System Version: " /tmp/tideway-hw-$$. | sed 's/^.*: Mac OS X \(\.*\)/os: Mac OS X \1/
rm -f /tmp/tideway-hw-$$. 
echo 'os_arch:' `uname -p 2>/dev/null`
```

is replaced with:

```bash
echo "hostname:" `hostname -s 2>/dev/null`
system_profiler SPHardwareDataType SPSoftwareDataType > /tmp/tideway-hw-$$. 2>/dev/null
grep "Computer Name: " /tmp/tideway-hw-$$. | sed 's/^.*: \(.*\)/description: \1/'
if [ -r /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf`
fi
grep "System Version: " /tmp/tideway-hw-$$. | sed 's/^.*:/os:/'
rn -f /tmp/tideway-hw-$$. 
echo 'os_arch:' `uname -p 2>/dev/null`
```

getFileSystems

The following code:

```bash
echo begin df:
df -lk 2>/dev/null 
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then 
cat /etc/xtab
fi
echo end xtab:
```

is replaced with:

```bash
echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then 
cat /etc/xtab
fi
echo end xtab:
```
echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:

The following code:

```
# for IRIX

configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```

is replaced with:

```
# for IRIX

configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```

IRIX

The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses
initialise

The following code:

```bash
PRIV_LS() {
    ls "$@
}
```

is replaced with:

```bash
PRIV_LS() {
    ls "$@
}

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
    "$@`
}
```

getNetworkConnectionList

The following code:

```bash
netstat -an -f inet 2>/dev/null
```

is replaced with:

```bash
netstat -an -f inet 2>/dev/null
netstat -anW -f inet6 2>/dev/null
```

getDeviceInfo

The following code:

```bash
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -s 2>/dev/null` `uname -R | awk '{print $2;}'`
ihn=`hostname 2>/dev/null`
  echo 'hostname:' $ihn
if [-r /etc/resolv.conf ]; then
  echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
  echo 'os:' `uname -s 2>/dev/null` `uname -R | awk '{print $2;}'`

getFileSystems

The following code:

echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [-f /etc/xtab ]; then
  cat /etc/xtab
fi
echo end xtab:

is replaced with:

echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [-r /etc/xtab ]; then
  cat /etc/xtab
fi
echo end xtab:

The following code:
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
    if [ -f $configfile ]; then
        cat $configfile
    fi
fi

is replaced with:

```
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
    if [ -r $configfile ]; then
        cat $configfile
    fi
fi
```

### POWER HMC

The following method:
The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses

The `netstat` command is replaced by `hostinfo` in the `getProcessList` method.
The `lshmc-n` command is replaced by `monhmc-rdisk` in the `getPatchList` method.

### Tru64

The following method:
The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses
**getHostInfo**

The following code:

```bash
echo 'kernel:' `uname -v 2>/dev/null`
echo 'model:' `hwmgr get attr -cat platform -a name 2>/dev/null | grep name | sed 's/name =//'`
echo 'begin tru64_psrinfo:'
  psrinfo -v
echo 'end tru64_psrinfo'
```

is replaced with:

```bash
echo 'kernel:' `uname -v 2>/dev/null'
echo 'model:' `PRIV_HWMGR hwmgr get attr -cat platform -a name 2>/dev/null | grep name | s
vmstat -P 2>/dev/null | grep "^Total Physical Memory" | sed -e 's/Total Physical Memory *= */s
/sbin/consvar -g sys_serial_num 2>/dev/null | sed -e 's/sys_serial_num *= */serial: /'
echo 'begin tru64_psrinfo:'
  psrinfo -v
echo 'end tru64_psrinfo'
```

**initialise**

The following code:

```bash
ls "$@"
```

```bash
# setld requires superuser privileges to display information on packages
PRIV_SETLD() {
  "$@
}
```

is replaced with:

```bash
ls "$@"
```

```bash
# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
  "$@
}
```
getDeviceInfo
The following code:

```
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

is replaced with:

```
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -r /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

getFileSystems
The following code:

```
echo begin df:
df -k -t nonfs,nfsv3 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
```
is replaced with:

```bash
echo begin df:
PRIV_DF df -k -t nonfs,nfsv3 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
```

The following code:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
cat $configfile
fi
fi
```

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
cat $configfile
fi
fi
```

OpenVMS

The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- `getNetworkInterfaces`
- `getIPAddresses`

**Mainframe**

The `getDatabaseDetail` method is now disabled by default. Previously it was enabled.

**VMware ESXi**

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
- `getNetworkInterfaces`
- `getIPAddresses`

**getHostInfo**

The following code:

```bash
/sbin/smbiosDump | awk '{ if( $1 ~ /Serial:/ ) { sub(".*Serial: *=",""); gsub( "\", ""); p} }' | head -1
echo begin vim-cmd-hostsummary:
vim-cmd hostsvc/hostsummary
```

is replaced with:

```bash
#!/bin/bash

# Get the Serial number from System Info block - Each block starts with double space and [A-Za-z]
serial=`/sbin/smbiosDump | sed -n '/^  System Info:/,/^  [A-Za-z]/ p' | grep '^ *Serial:'`

# If the Serial number does NOT exist in the System block, get it from Board Info
if [ "serial" = "" ]; then
    serial=`/sbin/smbiosDump | sed -n '/^  Board Info:/,/^  [A-Za-z]/ p' | grep '^ *Serial:'`
fi

if [ "serial" != "" ]; then
    echo serial: `echo $serial | cut -d: -f2- | sed 's/"//g'`
fi

echo begin vim-cmd-hostsummary:
vim-cmd hostsvc/hostsummary
```
**initialise**

The following code:

```
PRIV_LS() {
    ls "@"
}
```

is replaced with:

```
PRIV_LS() {
    ls "@"
}
```

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
    "@"
}

**getDeviceInfo**

The following code:

```
echo 'hostname:' $ihn
echo 'fqdn:' `hostname -f 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "dns_domain" = "" -a -r /etc/resolv.conf ]; then
    dns_domain=`awk '/^domain/ {sub(/\000/, "", $2); print $2; exit }' /etc/resolv.conf`
fi
echo 'dns_domain: ' $dns_domain
```

is replaced with:

```
echo 'hostname:' $ihn
echo 'fqdn:' `hostname -f 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "dns_domain" = "" -a -r /etc/resolv.conf ]; then
    dns_domain=`awk '/^domain/ {sub(/\000/, "", $2); print $2; exit }' /etc/resolv.conf`
fi
echo 'dns_domain: ' $dns_domain
```
**getFileSystems**

The following code:

```bash
echo begin df:
df -k 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
```

is replaced with:

```bash
echo begin df:
PRIV_DF df -k 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
```

**UnixWare**

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
- `getNetworkInterfaces`
- `getIPAddresses`

**initialise**

The following code is added:

```bash
# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
    "$@
}
```

**getDeviceInfo**

The following code:
```
echo 'hostname:' `uname -n 2>/dev/null`
if [-f /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sv 2>/dev/null`
```

is replaced with:

```
echo 'hostname:' `uname -n 2>/dev/null`
if [-r /etc/resolv.conf ]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sv 2>/dev/null`
```

**getFileSystems**

The following code:

```
getFileSizeSystems
```

```
echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
if [-f /etc/mnttab ]; then
cat /etc/mnttab
fi
echo end mount:
echo begin xtab:
if [-f /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
```

is replaced with:

```
echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
echo begin mount:
if [-r /etc/mnttab ]; then
cat /etc/mnttab
fi
echo end mount:
echo begin xtab:
if [-r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
```
The following code:

```bash
fi
echo end xtab:
```

is replaced with:

```bash
fi
```

**FreeBSD**

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
- `getNetworkInterfaces`
- `getIPAddresses`

`getNetworkConnectionList`

The following code is added:

```bash
netstat -anf inet6 -W 2>/dev/null
```
initialise

The following code is added:

```bash
# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
  "$@
}
```

getDeviceInfo

The following code:

```bash
ihn=localhost
fi
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
  echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

is replaced with:

```bash
ihn=localhost
fi
echo 'hostname:' $ihn
if [ -r /etc/resolv.conf ]; then
  echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

getFileSystems

The following code:

```bash
echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
```
```bash
if [ -f /etc/xtab ]; then
cat /etc/xtab
fi
```

is replaced with:

```bash
echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
```

```bash
if [ -r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
```

The following code:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
    cat $configfile
  fi
fi
```

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```

**Linux**

The following method:
* getInterfaceList

is replaced with:

* getMACAddresses
* getNetworkInterfaces
* getIPAddresses

**getHostInfo**

The following code:

```bash
if [ -f /usr/sbin/esxcfg-info ]; then
  # On a VMware ESX controller, report the *real* hardware information
  file=/tmp/tideway-hw-$$
  PRIV_ESXCFG /usr/sbin/esxcfg-info --hardware > ${file} 2>/dev/null
  uuid=""
  if [ $? -eq 0 ]; then
    physical=`grep "Num Packages\." ${file} | sed -e "s/[\^0-9]//g"`
    logical=`grep "Num Cores." ${file} | head -n 1 | sed -e "s/[\^0-9]//g"
```

is replaced with:

```bash
if [ -f /usr/sbin/esxcfg-info ]; then
  # On a VMware ESX controller, report the *real* hardware information
  file=/tmp/tideway-hw-$$
  uuid=""
  PRIV_ESXCFG /usr/sbin/esxcfg-info --hardware > ${file} 2>/dev/null
  if [ $? -eq 0 ]; then
    physical=`grep "Num Packages\." ${file} | sed -e "s/[\^0-9]//g"
    logical=`grep "Num Cores." ${file} | head -n 1 | sed -e "s/[\^0-9]//g"
```

The following code:

```bash
# zLinux?
if [ -f /proc/sysinfo -a -d /proc/dasd ]; then
  echo "candidate_vendor[]:" 'egrep 'Manufacturer:' /proc/sysinfo | awk '{print $2;}''
  type=`egrep '^Type:' /proc/sysinfo | awk '{print $2;}'`
  model=`egrep '^Model:' /proc/sysinfo | awk '{print $2;}'`
```

is replaced with:

```bash
```
# zLinux?
if [ -r /proc/sysinfo -a -d /proc/dasd ]; then
  echo "candidate_vendor[]:" `egrep '^Manufacturer:' /proc/sysinfo | awk '{print $2;}'`
  type=`egrep '^Type:' /proc/sysinfo | awk '{print $2;}'
  model=`egrep '^Model:' /proc/sysinfo | awk '{print $2;}'
getPackageList
The following code:

COLUMNS=256 dpkg -l "*" | egrep '^ii '

is replaced with:

COLUMNS=256 dpkg -l "*" 2>/dev/null | egrep '^ii '

initialise
The following code is added:

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
  "$@
}

getDeviceInfo
The following code:

echo 'hostname:' $ihn
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" -a -f /etc/resolv.conf ]; then
dns_domain=`awk '/(domain|search)/ {sub(/\\000/, ",", $2); print $2; exit }' /etc/resolv.conf`
echo 'dns_domain: ' $dns_domain

is replaced with:

echo 'hostname:' $ihn
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" -a -f /etc/resolv.conf ]; then
dns_domain=`awk '/(domain|search)/ {sub(/\\000/, ",", $2); print $2; exit }' /etc/resolv.conf`
echo 'dns_domain: ' $dns_domain
The following code:

```
# SuSE lsb_release does provide service pack so prefer SuSE-release file.
o $os = "" -a -f /etc/SuSE-release "; then
  os=`cat /etc/SuSE-release`
fi
if [ "$os" = "" -a -x /usr/bin/lsb_release ]; then
  # We'd like to use -ds but that puts quotes in the output!
o $os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed 's/^[ ]\{1\}\(/\( )/'
  if [ "$os" = "(none)" ]; then
    os=""
  else
    # Check to see if its a variant of Red Hat
    rpm -q oracle-logos >/dev/null 2>&1
    if [ $? -eq 0 ]; then
      # Oracle variant
      os="Oracle $os"
    fi
  fi
if [ "$os" = "" -a -r /proc/vmware/version ]; then
  os=`grep -m1 ESX /proc/vmware/version`
fi
if [ "$os" = "" -a -r /etc/esx-release ]; then
  os=`grep ESX /etc/esx-release`
fi
if [ "$os" = "" -a -r /etc/redhat-release ]; then
  os=`cat /etc/redhat-release`
```

is replaced with:

```
# SuSE lsb_release does provide service pack so prefer SuSE-release file.
o $os = "" -a -f /etc/SuSE-release "; then
  os=`cat /etc/SuSE-release`
fi
if [ "$os" = "" -a -x /usr/bin/lsb_release ]; then
  # We'd like to use -ds but that puts quotes in the output!
o $os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed 's/^[ ]\{1\}\(/\( )/'
  if [ "$os" = "(none)" ]; then
    os=""
  else
    # Check to see if its a variant of Red Hat
    rpm -q oracle-logos >/dev/null 2>&1
    if [ $? -eq 0 ]; then
      # Oracle variant
      os="Oracle $os"
    fi
  fi
fi
if [ "$os" = "" -a -r /proc/vmware/version ]; then
  os=`grep -m1 ESX /proc/vmware/version`
fi
if [ "$os" = "" -a -r /etc/vmware-release ]; then
  os=`grep ESX /etc/vmware-release`
fi
if [ "$os" = "" -a -r /etc/redhat-release ]; then
  os=`cat /etc/redhat-release`
```
if [ "$os" = "" -a -x /usr/bin/lsb_release ]; then
    # We'd like to use -ds but that puts quotes in the output!
    os=`/usr/bin/lsb_release -d | cut -f2 -d:`
fi
if [ "$os" = "" -a -e /proc/vmware/version ]; then
    os=`grep -m1 ESX /proc/vmware/version`
fi
if [ "$os" = "" -a -f /etc/vmware-release ]; then
    os=`grep ESX /etc/vmware-release`
fi
if [ "$os" = "" -a -f /etc/SuSE-release ]; then
    os=`head -n 1 /etc/SuSE-release`
fi
if [ "$os" = "" -a -f /etc/debian_version ]; then
    ver=`cat /etc/debian_version`
    os="Debian Linux $ver"
fi
if [ "$os" = "" -a -f /etc/mandrake-release ]; then
    os=`cat /etc/mandrake-release`
fi
if [ "$os" = "" ]; then
    is replaced with:

    if [ "$os" = "" -a -r /etc/debian_version ]; then
        ver=`cat /etc/debian_version`
        os="Debian Linux $ver"
    fi
    if [ "$os" = "" -a -r /etc/mandrake-release ]; then
        os=`cat /etc/mandrake-release`
    fi
    if [ "$os" = "" ]; then

getFileSystems

The following code:

echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null

is replaced with:

```bash
echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
```
The following code:

```bash
echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
```

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
    if [ -f $configfile ]; then
        cat $configfile
    fi
fi
```

**OpenBSD**

The following method:

- getInterfaceList

is replaced with:

- getMACAddresses
- getNetworkInterfaces
- getIPAddresses

**initialise**

The following code is added:

```bash
# This function supports privilege listing of file systems and related
```
# size and usage.
PRIV_DF() {
    "$@
}

getNetworkConnectionList
The following code:

```
netstat -an -f inet -W 2>/dev/null
```

is replaced with:

```
netstat -an -f inet -v 2>/dev/null
netstat -an -f inet6 -v 2>/dev/null
```

getDeviceInfo
The following code:

```
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [-f /etc/resolv.conf]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

is replaced with:

```
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [-r /etc/resolv.conf]; then
    echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'os:' `uname -sr 2>/dev/null`
```

getFileSystems
The following code:
```
echo begin df:
df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
n
is replaced with:

echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:
echo begin mount:
mount 2>/dev/null
echo end mount:
echo begin xtab:
if [ -r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
n
The following code:

```
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
    cat $configfile
  fi
fi
```

is replaced with:

```
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```
**NetBSD**

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
- `getNetworkInterfaces`
- `getIPAddresses`

**initialise**

The following code is added:

```bash
# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
    "$@
}
```

**getNetworkConnectionList**

The following code:

```bash
netstat -an -f inet -W 2>/dev/null
```

is replaced with:

```bash
netstat -an -f inet -v 2>/dev/null
netstat -an -f inet6 -v 2>/dev/null
```

**getDeviceInfo**

The following code:

```bash
ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
```
echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi

echo 'os:' `uname -sr 2>/dev/null`

is replaced with:

ihn=`hostname 2>/dev/null`

echo 'hostname:' $ihn

if [ -r /etc/resolv.conf ]; then
  echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi

echo 'os:' `uname -sr 2>/dev/null`

getFileSystem

echo begin df:
df -lk 2>/dev/null
echo end df:

is replaced with:

echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:

The following code:
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
    cat $configfile
  fi
fi

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi```

**HP-UX**

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
- `getNetworkInterfaces`
- `getIPAddresses`
- `getNetworkConnectionList`

The following code is added:

```
netstat -an -f inet6 2>/dev/null
```

**getHostInfo**

The following code:

```
fi
echo 'license:' `uname -l`
# machinfo indicates Itanium based system
```
if [ -x /usr/contrib/bin/machinfo ]; then
    echo 'begin machinfo:'
    if [ $OSVER -ge 1131 ]; then
        fi
        echo 'license: `uname -l`
    fi
if [ -x /usr/contrib/bin/machinfo ]; then
    echo 'begin machinfo:'
    if [ $OSVER -ge 1131 ]; then
        fi
        cores=`/usr/sbin/ioscan -kC processor 2>/dev/null | grep processor | wc -l`
        fi
if [ "$OSVER" -ge 1100 ]; then
    typeset -i2 bin
    bin=`getconf CPU_CHIP_TYPE`
    typeset -i16 hex
    hex=2#`echo $bin | sed -e 's/2#//' -e 's/.....$//'`
    chip_type=`echo $hex | cut -c4-`
    fi
    if [ "$chip_type" = "20" ]; then
        cores_per_processor=2
    else
        cores_per_processor=1
    fi
is replaced with:

cores=`/usr/sbin/ioscan -kC processor 2>/dev/null | grep processor | wc -l`
fi
if [ "$OSVER" -ge 1100 ]; then
    chip_type=$((`getconf CPU_CHIP_TYPE` >> 5))
    fi
if [ "$chip_type" = "20" ]; then
    cores_per_processor=2
else
    cores_per_processor=1
The following code:

```bash
echo "cpu_chip_type:" $chip_type
fi
echo 'cpu_model:' `model 2>/dev/null`
KERNEL=/stand/vmunix
MEMDEV=/dev/mem
if [ -r $MEMDEV ]; then
  if [ "$kernel_bits" -eq 64 ]; then
    ADB="adb64 -k"
  else
    ADB="adb -k"
  fi
  echo itick_per_usec/D | $ADB $KERNEL $MEMDEV 2>/dev/null | egrep -v ':$'
fi
echo "end hpux_cpu_info"
if [ $ram -eq 0 ]; then
```

is replaced with:

```bash
echo "cpu_chip_type:" $chip_type
fi
echo 'cpu_model:' `model 2>/dev/null`
echo "end hpux_cpu_info"
if [ $ram -eq 0 ]; then
```

getPackageList

The following code:

```bash
PRIV_SWLIST swlist -l product 2>/dev/null | grep -v '^#|PH[A-Z]{2}[0-9]+'
```

is replaced with:

```bash
PRIV_SWLIST swlist -v -l product -a tag -a revision -a title -a vendor_tag 2>/dev/null
```

initialise

The following code is added:
# This function supports privilege listing of file systems and related size and usage.

PRIV_DF() { 
   "$@
}

getDeviceInfo

The following code:

ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -f /etc/resolv.conf ]; then
   echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`

is replaced with:

ihn=`hostname 2>/dev/null`
echo 'hostname:' $ihn
if [ -r /etc/resolv.conf ]; then
   echo 'dns_domain:' `awk '/^(domain|search)/ { print $2; exit }' /etc/resolv.conf 2>/dev/null`
fi
echo 'domain:' `domainname 2>/dev/null`

getFileSystems

The following code:

echo begin df:
bdf -l 2>/dev/null
echo end df:
echo begin mount:
mount -p 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:

is replaced with:

echo begin df:
bdf -l 2>/dev/null
echo end df:
echo begin mount:
mount -p 2>/dev/null
echo end mount:
echo begin xtab:
if [ -f /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:
echo begin df:
PRIV_DF bdf -l 2>/dev/null
echo end df:
echo begin mount:
mount -p 2>/dev/null
echo end mount:
echo begin xtab:
if [ -r /etc/xtab ]; then
cat /etc/xtab
fi
echo end xtab:

The following code:

echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
    cat $configfile
  fi
fi

is replaced with:

```bash
echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'`
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi
```

VMware ESX

The PATH environment variable has changed from `/bin:/sbin` to `/bin:/usr/bin:/sbin:
/usr/sbin`.

The following method:

- `getInterfaceList`

is replaced with:

- `getMACAddresses`
getNetworkInterfaces

getIPAddresses

**getHostInfo**

The following code:

```bash
if [ -f /usr/sbin/esxcfg-info ]; then
    # On a VMWare ESX controller, report the *real* hardware information
    file=/tmp/tideway-hw-$$
    PRIV_ESXCFG /usr/sbin/esxcfg-info --hardware > ${file} 2>/dev/null
    uuid=""
    if [ $? -eq 0 ]; then
        physical=`grep "Num Packages." ${file} | sed -e "s/[^0-9]//g"`
        logical=`grep "Num Cores." ${file} | head -n 1 | sed -e "s/[^0-9]//g"
    fi
fi
```

is replaced with:

```bash
if [ -f /usr/sbin/esxcfg-info ]; then
    # On a VMWare ESX controller, report the *real* hardware information
    file=/tmp/tideway-hw-$$
    uuid=""
    PRIV_ESXCFG /usr/sbin/esxcfg-info --hardware > ${file} 2>/dev/null
    if [ $? -eq 0 ]; then
        physical=`grep "Num Packages." ${file} | sed -e "s/[^0-9]//g"`
        logical=`grep "Num Cores." ${file} | head -n 1 | sed -e "s/[^0-9]//g"
    fi
fi
```

The following code:

```bash
# zLinux?
if [ -f /proc/sysinfo -a -d /proc/dasd ]; then
    echo "candidate_vendor[]:" `egrep '^Manufacturer:' /proc/sysinfo | awk '{print $2;}'`
    type=`egrep '^Type:' /proc/sysinfo | awk '{print $2;}'`
    model=`egrep '^Model:' /proc/sysinfo | awk '{print $2;}'`
fi
```

is replaced with:

```bash
# zLinux?
if [ -r /proc/sysinfo -a -d /proc/dasd ]; then
    echo "candidate_vendor[]:" `egrep '^Manufacturer:' /proc/sysinfo | awk '{print $2;}'`
    type=`egrep '^Type:' /proc/sysinfo | awk '{print $2;}'`
    model=`egrep '^Model:' /proc/sysinfo | awk '{print $2;}'`
fi
```
Documentation

BMC Software Confidential

getPackageList
The following code:

COLUMNS=256 dpkg -l '*' | egrep '^ii '

is replaced with:

COLUMNS=256 dpkg -l '*' 2>/dev/null | egrep '^ii '

initialise
The following code is added:

# This function supports privilege listing of file systems and related
# size and usage.
PRIV_DF() {
"$@"
}

getDeviceInfo
The following code:

echo 'hostname:' $ihn
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" -a -f /etc/resolv.conf ]; then
dns_domain=`awk '/^(domain|search)/ {sub(/\\\\000$/, "", $2); print $2; exit }' /etc/resolv.conf 2>/dev/nu
fi
echo 'dns_domain: ' $dns_domain

is replaced with:

echo 'hostname:' $ihn
echo 'fqdn:' `hostname --fqdn 2>/dev/null`
dns_domain=`hostname -d 2>/dev/null | sed -e 's/(none)//'`
if [ "$dns_domain" = "" -a -r /etc/resolv.conf ]; then

BMC Atrium Discovery 10.2

Page 110 of 1671


Documentation

BMC Software Confidential

dns_domain=`awk '/^(domain|search)/ {sub(/\\\\000$/, "", $2); print $2; exit }' /etc/resolv.conf 2>/dev/nu
fi
echo 'dns_domain: ' $dns_domain

The following code:

os=""
if [ "$os" = "" -a -f /etc/redhat-release ]; then

is replaced with:

os=""
# SuSE lsb_release does provide service pack so prefer SuSE-release file.
if [ "$os" = "" -a -r /etc/SuSE-release ]; then
os=`cat /etc/SuSE-release`
fi
if [ "$os" = "" -a -x /usr/bin/lsb_release ]; then
# We'd like to use -ds but that puts quotes in the output!
os=`/usr/bin/lsb_release -d | cut -f2 -d: | sed -e 's/^[ \t]//'`
if [ "$os" = "(none)" ]; then
os=""
else
# Check to see if its a variant of Red Hat
rpm -q oracle-logos > /dev/null 2>&1
if [ $? -eq 0 ]; then
# Oracle variant
os="Oracle $os"
fi
fi
fi
if [ "$os" = "" -a -r /proc/vmware/version ]; then
os=`grep -m1 ESX /proc/vmware/version`
fi
if [ "$os" = "" -a -r /etc/vmware-release ]; then
os=`grep ESX /etc/vmware-release`
fi
if [ "$os" = "" -a -r /etc/redhat-release ]; then
os=`cat /etc/redhat-release`

The following code:

os="Oracle $os"
fi
fi
if [ "$os" = "" -a -x /usr/bin/lsb_release ]; then

BMC Atrium Discovery 10.2

Page 111 of 1671


# We'd like to use -ds but that puts quotes in the output!
    os=`/usr/bin/lsb_release -d | cut -f2 -d:`
fi
if [ "$os" = "" -a -e /proc/vmware/version ]; then
    os=`grep -m1 ESX /proc/vmware/version`
fi
if [ "$os" = "" -a -f /etc/vmware-release ]; then
    os=`grep ESX /etc/vmware-release`
fi
if [ "$os" = "" -a -f /etc/SuSE-release ]; then
    os=`head -n 1 /etc/SuSE-release`
fi
if [ "$os" = "" -a -f /etc/debian_version ]; then
    ver=`cat /etc/debian_version`
    os="Debian Linux $ver"
fi
if [ "$os" = "" -a -r /etc/debian_version ]; then
    ver=`cat /etc/debian_version`
    os="Debian Linux $ver"
fi
if [ "$os" = "" -a -r /etc/mandrake-release ]; then
    os=`cat /etc/mandrake-release`
fi
if [ "$os" = "" ]; then

is replaced with:

    os="Oracle $os"
fi
fi
if [ "$os" = "" -a -f /etc/debian_version ]; then
    ver=`cat /etc/debian_version`
    os="Debian Linux $ver"
fi
if [ "$os" = "" -a -r /etc/mandrake-release ]; then
    os=`cat /etc/mandrake-release`
fi
if [ "$os" = "" ]; then

getFileSystems

The following code:

    echo begin df:
    df -lk 2>/dev/null
    echo end df:

is replaced with:

    echo begin df:
    df -lk 2>/dev/null
    echo end df:
echo begin df:
PRIV_DF df -lk 2>/dev/null
echo end df:

The following code:

echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'
if [ "$configfile" != "" ]; then
  if [ -f $configfile ]; then
    cat $configfile
  fi
fi

is replaced with:

echo begin smbconf:
configfile=`smbstatus -v 2>/dev/null | grep "using configfile" | awk '{print $4;}'
if [ "$configfile" != "" ]; then
  if [ -r $configfile ]; then
    cat $configfile
  fi
fi

Package list - BMC Atrium Discovery 10.2

The following table provides a list of all the OS and application packages contained in a default BMC Atrium Discovery 10.2 Enterprise Edition installation:

<table>
<thead>
<tr>
<th>Package name</th>
<th>Package name</th>
<th>Package name</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl-2.2.49-6.el6.x86_64</td>
<td>apr-1.3.9-5.el6_2.x86_64</td>
<td>apr-devel-1.3.9-5.el6_2.x86_64</td>
</tr>
<tr>
<td>apr-util-1.3.9-3.el6_0.1.x86_64</td>
<td>apr-util-devel-1.3.9-3.el6_0.1.x86_64</td>
<td>apr-util-mdap-1.3.9-3.el6_0.1.x86_64</td>
</tr>
<tr>
<td>attr-2.4.44-7.el6.x86_64</td>
<td>audit-2.3.7-5.el6.x86_64</td>
<td>audit-libs-2.3.7-5.el6.x86_64</td>
</tr>
<tr>
<td>basesystem-10.0-0.el6.noarch</td>
<td>bash-4.1.2-29.el6.x86_64</td>
<td>bc-1.06.95-1.el6.x86_64</td>
</tr>
<tr>
<td>bind-libs-9.8.2-0.30.rc1.el6_6.2.x86_64</td>
<td>bind-utils-9.8.2-0.30.rc1.el6_6.2.x86_64</td>
<td>binutils-2.20.51.0.2-5.42.el6.x86_64</td>
</tr>
<tr>
<td>busybox-1.15.1-21.el6_6.x86_64</td>
<td>bzip2-1.0.5-7.el6_0.x86_64</td>
<td>bzip2-libs-1.0.5-7.el6_0.x86_64</td>
</tr>
<tr>
<td>ca-certificates-2015.2.4-65.0.1.el6_6. noarch</td>
<td>checkpolicy-2.0.22-1.el6.x86_64</td>
<td>chkconfig-1.3.49.3-5.el6.x86_64</td>
</tr>
<tr>
<td>cifs-utils-4.8.1-19.el6.x86_64</td>
<td>cloog-ppl-0.15.7-1.2.el6.x86_64</td>
<td>compat-libstdc++-33-3.2.3-69.el6.x86_64</td>
</tr>
<tr>
<td>coreutils-8.4-37.el6.x86_64</td>
<td>coreutils-libs-8.4-37.el6.x86_64</td>
<td>cpio-2.10-12.el6_5.x86_64</td>
</tr>
<tr>
<td>Package name</td>
<td>Package name</td>
<td>Package name</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>cpp-4.4.7-11.el6.x86_64</td>
<td>cracklib-2.8.16-4.el6.x86_64</td>
<td>cracklib-dicts-2.8.16-4.el6.x86_64</td>
</tr>
<tr>
<td>cronie-1.4.4-15.el6.x86_64</td>
<td>cronie-anacron-1.4.4-15.el6.x86_64</td>
<td>crontabs-1.10-33.el6.noarch</td>
</tr>
<tr>
<td>curl-7.19.7-40.el6_6.4.x86_64</td>
<td>cyrus-sasl-2.1.23-15.el6_6.2.x86_64</td>
<td>cyrus-sasl-devel-2.1.23-15.el6_6.2.x86_64</td>
</tr>
<tr>
<td>db4-cxx-4.7.25-18.el6_4.x86_64</td>
<td>db4-devel-4.7.25-18.el6_4.x86_64</td>
<td>db4-utils-4.7.25-18.el6_4.x86_64</td>
</tr>
<tr>
<td>device-mapper-event-1.02.90-2.el6_6.2.x86_64</td>
<td>device-mapper-event-libs-1.02.90-2.el6_6.2.x86_64</td>
<td>device-mapper-lib-1.02.90-2.el6_6.2.x86_64</td>
</tr>
<tr>
<td>device-mapper-multipath-0.4.9-80.el6_6.3.x86_64</td>
<td>device-mapper-multipath-libs-0.4.9-80.el6_6.3.x86_64</td>
<td>device-mapper-persistent-data-0.3.2.1.el6.x86_64</td>
</tr>
<tr>
<td>dhclient-4.1.1-43.P1.el6_6.1.x86_64</td>
<td>dhcp-common-4.1.1-43.P1.el6_6.1.x86_64</td>
<td>diffutils-2.8.1-28.el6.x86_64</td>
</tr>
<tr>
<td>dmidecode-2.12-5.el6_5.x86_64</td>
<td>dos2unix-3.1-37.el6.x86_64</td>
<td>dracut-004-356.el6_6.2.noarch</td>
</tr>
<tr>
<td>e2fsprogs-1.41.12-21.el6.x86_64</td>
<td>e2fsprogs-libs-1.41.12-21.el6.x86_64</td>
<td>elfutils-libelf-0.158-3.2.el6.x86_64</td>
</tr>
<tr>
<td>expat-2.0.1-11.el6_2.x86_64</td>
<td>file-5.04-21.el6.x86_64</td>
<td>file-lib-5.04-21.el6.x86_64</td>
</tr>
<tr>
<td>filesystem-2.4.30-3.el6.x86_64</td>
<td>findutils-4.4.2-6.el6.x86_64</td>
<td>fipscheck-1.2.0-7.el6.x86_64</td>
</tr>
<tr>
<td>grep-2.6.3-6.el6.x86_64</td>
<td>grubby-7.0.15-7.el6.x86_64</td>
<td>gmp-4.3.1-7.el6.x86_64</td>
</tr>
<tr>
<td>httpd-2.2.15-39.el6.x86_64</td>
<td>hwdata-0.233-11.1.el6.noarch</td>
<td>gnutls-2.8.5-14.el6_6.5.x86_64</td>
</tr>
<tr>
<td>iptables-ipv6-1.4.7-14.el6.x86_64</td>
<td>iptables-1.4.7-14.el6.x86_64</td>
<td>gpg-pubkey-66a46459-fffe350</td>
</tr>
<tr>
<td>kernel-2.6.32-504.16.2.el6.x86_64</td>
<td>kernel-headers-2.6.32-504.16.2.el6.x86_64</td>
<td>gpg-pubkey-66a46459-fffe350</td>
</tr>
<tr>
<td>Package name</td>
<td>Package name</td>
<td>Package name</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>------------------------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>keyutils-1.4.5-1.el6.x86_64</td>
<td>keyutils-libs-1.4.5-1.el6.x86_64</td>
<td>kpartx-0.4.9-80.el6_6.3.x86_64</td>
</tr>
<tr>
<td>krb5-appl-clients-1.0.1-7.el6_2.1.x86_64</td>
<td>krb5-libs-1.10.3-37.el6_6.x86_64</td>
<td>krb5-workstation-1.10.3-37.el6_6.x86_64</td>
</tr>
<tr>
<td>less-436-13.el6.x86_64</td>
<td>libacl-2.2.49-6.el6.x86_64</td>
<td>libaio-0.3.107-10.el6.x86_64</td>
</tr>
<tr>
<td>libattr-2.4.47-7.el6.x86_64</td>
<td>libblkid-2.17.2-12.18.el6.x86_64</td>
<td>libcap-2.16-5.5.el6.x86_64</td>
</tr>
<tr>
<td>libcap-ng-0.6.4-3.el6.x86_64</td>
<td>libcom_err-1.41.12-21.el6.x86_64</td>
<td>libcurl-7.19.7-40.el6_6.4.x86_64</td>
</tr>
<tr>
<td>libdrm-2.4.52-4.el6.x86_64</td>
<td>libedit-2.11-4.20080712cvs.1.el6.x86_64</td>
<td>libffi-3.0.5-3.2.el6.x86_64</td>
</tr>
<tr>
<td>libgcc-4.4.7-11.el6.x86_64</td>
<td>libgcrypt-1.4.5-11.el6.x86_64</td>
<td>libgomp-4.4.7-11.el6.x86_64</td>
</tr>
<tr>
<td>libgpg-error-1.7-4.el6.x86_64</td>
<td>libgssglue-0.1-11.el6.x86_64</td>
<td>libicu-4.2.1-11.el6.x86_64</td>
</tr>
<tr>
<td>libidn-1.18-2.el6.x86_64</td>
<td>libnih-1.0.1-7.el6.x86_64</td>
<td>libgssglue-0.1-11.el6.x86_64</td>
</tr>
<tr>
<td>libpciaccess-0.13.3-0.1.el6.x86_64</td>
<td>libselinux-2.0.94-5.8.el6.x86_64</td>
<td>libgssglue-0.1-11.el6.x86_64</td>
</tr>
<tr>
<td>libsemanage-2.0.43-4.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
<td>libidn-1.18-2.el6.x86_64</td>
</tr>
<tr>
<td>libssh2-1.4.2-1.el6_6.1.x86_64</td>
<td>libstdc++-4.4.7-11.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
</tr>
<tr>
<td>libtasn1-2.3-6.el6_5.x86_64</td>
<td>libtevent-0.9.18-3.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
</tr>
<tr>
<td>libtoolc-0.2.1-10.el6.x86_64</td>
<td>libtevent-0.9.18-3.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
</tr>
<tr>
<td>libuser-0.56.13-5.el6.x86_64</td>
<td>libvirt-1.1.5-4.1.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
</tr>
<tr>
<td>libxml2-2.7.6-17.el6_6.1.x86_64</td>
<td>libvirt-1.1.5-4.1.el6.x86_64</td>
<td>libsepol-2.0.41-4.el6.x86_64</td>
</tr>
<tr>
<td>logrotate-3.7.8-17.el6.x86_64</td>
<td>lua-5.1.4-4.1.el6.x86_64</td>
<td>libvirt-1.1.5-4.1.el6.x86_64</td>
</tr>
<tr>
<td>lvm2-libs-2.0.2.111-2.el6.x6.2.x86_64</td>
<td>lvm2-2.02.111-2.el6_6.2.x86_64</td>
<td>lvm2-2.02.111-2.el6_6.2.x86_64</td>
</tr>
<tr>
<td>mailcap-2.1.31-2.el6.noarch</td>
<td>lzo-2.0-3.1.el6_5.x86_64</td>
<td>lvm2-2.02.111-2.el6_6.2.x86_64</td>
</tr>
<tr>
<td>MAKEDEV-3.24-6.el6.x86_64</td>
<td>mailx-12.4-8.el6.x86_64</td>
<td>m4-1.4.13-5.el6.x86_64</td>
</tr>
<tr>
<td>mingetty-1.0.8-5.el6.x86_64</td>
<td>make-3.81-20.el6.x86_64</td>
<td>make-3.81-20.el6.x86_64</td>
</tr>
<tr>
<td>module-init-tools-3.9-24.el6.x86_64</td>
<td>mysql-libs-5.1.73-3.el6_5.x86_64</td>
<td>mysql-libs-5.1.73-3.el6_5.x86_64</td>
</tr>
<tr>
<td>nano-2.0.9-7.el6.x86_64</td>
<td>ncurses-5.7-4.20090207.el6.x86_64</td>
<td>ncurses-5.7-4.20090207.el6.x86_64</td>
</tr>
<tr>
<td>ncurses-libs-5.7-4.20090207.el6.x86_64</td>
<td>net-snmp-5.5-50.el6_6.1.x86_64</td>
<td>net-snmp-5.5-50.el6_6.1.x86_64</td>
</tr>
<tr>
<td>net-tools-1.60-110.el6_2.x86_64</td>
<td>nmap-6.47-1.rhel6.x86_64</td>
<td>nmap-6.47-1.rhel6.x86_64</td>
</tr>
<tr>
<td>nss-util-3.18.0-1.el6_6.x86_64</td>
<td>ntp-4.2.6p5-3.el6_6.x86_64</td>
<td>ntpdate-4.2.6p5-3.el6_6.x86_64</td>
</tr>
<tr>
<td>openssl-2.4.39-8.el6.x86_64</td>
<td>openssl-clients-2.4.39-8.el6.x86_64</td>
<td>openssl-devel-2.4.39-8.el6.x86_64</td>
</tr>
<tr>
<td>openssl-5.3p1-104.el6_6.1.x86_64</td>
<td>openssl-clients-5.3p1-104.el6_6.1.x86_64</td>
<td>openssl-clients-5.3p1-104.el6_6.1.x86_64</td>
</tr>
<tr>
<td>openssl-server-5.3p1-104.el6_6.1.x86_64</td>
<td>openssl-server-5.3p1-104.el6_6.1.x86_64</td>
<td>openssl-server-5.3p1-104.el6_6.1.x86_64</td>
</tr>
<tr>
<td>Package name</td>
<td>Package name</td>
<td>Package name</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>openssl-1.0.1e-30.el6_6.8.x86_64</td>
<td>p11-kit-0.18.5-2.el6_5.2.x86_64</td>
<td>p11-kit-trust-0.18.5-2.el6_5.2.x86_64</td>
</tr>
<tr>
<td>pam-1.1.1-20.el6.x86_64</td>
<td>parted-2.1-25.el6.x86_64</td>
<td>passwd-0.77-4.el6_2.2.x86_64</td>
</tr>
<tr>
<td>pcre-7.8-6.el6.x86_64</td>
<td>perl-5.10.1-136.el6_6.1.x86_64</td>
<td>perl-libs-5.10.1-136.el6_6.1.x86_64</td>
</tr>
<tr>
<td>perl-Module-Pluggable-3.90-136.el6_6.1. x86_64</td>
<td>perl-Pod-Escapes-1.04-136.el6_6.1. x86_64</td>
<td>perl-Pod-Simple-3.13-136.el6_6.1.x86_64</td>
</tr>
<tr>
<td>perl-version-0.77-136.el6_6.1.x86_64</td>
<td>pinentry-0.7.6-8.el6.x86_64</td>
<td>pkgconfig-0.23-9.1.el6.x86_64</td>
</tr>
<tr>
<td>plymouth-0.8.3-27.el6_5.1.x86_64</td>
<td>plymouth-core-libs-0.8.3-27.el6_5.1. x86_64</td>
<td>plymouth-scripts-0.8.3-27.el6_5.1.x86_64</td>
</tr>
<tr>
<td>policycoreutils-2.0.83-19.47.el6_6.1. x86_64</td>
<td>popt-1.13-7.el6.x86_64</td>
<td>postfix-2.6.6-6.el6_5.x86_64</td>
</tr>
<tr>
<td>ppp-0.10.2-11.el6.x86_64</td>
<td>procps-3.2.8-30.el6.x86_64</td>
<td>psmisc-22.6-19.el6_5.x86_64</td>
</tr>
<tr>
<td>pth-2.0.7-9.3.el6.x86_64</td>
<td>python-2.6.6-52.el6.x86_64</td>
<td>python-libs-2.6.6-52.el6.x86_64</td>
</tr>
<tr>
<td>readline-6.0-4.el6.x86_64</td>
<td>redhat-logos-60.0.14-1.el6.noarch</td>
<td>redhat-release-server-6Server-6.6.0.2.el6. x86_64</td>
</tr>
<tr>
<td>rootfiles-8.1-6.1.el6.noarch</td>
<td>rpcbind-0.2.0-11.el6.x86_64</td>
<td>rpm-4.8.0-38.el6_6.x86_64</td>
</tr>
<tr>
<td>rpm-libs-4.8.0-38.el6_6.x86_64</td>
<td>rsh-0.17-64.el6.x86_64</td>
<td>rsync-3.0.6-12.el6.x86_64</td>
</tr>
<tr>
<td>rsyslog-5.8.10-10.el6.x86_64</td>
<td>samba-client-3.6.23-14.el6.x86_64</td>
<td>samba-common-3.6.23-14.el6.x86_64</td>
</tr>
<tr>
<td>samba-winbind-3.6.23-14.el6.x86_64</td>
<td>samba-winbind-clients-3.6.23-14.el6.x86_64</td>
<td>screen-4.0.3-16.el6.x86_64</td>
</tr>
<tr>
<td>sed-4.2.1-10.el6.x86_64</td>
<td>selinux-policy-3.7.19-260.el6_6.3. noarch</td>
<td>selinux-policy-targeted-3.7.19-260.el6_6. 3.noarch</td>
</tr>
<tr>
<td>setup-2.8.14-20.el6_4.1.noarch</td>
<td>shadow-utils-4.1.4.2-19.el6_6.1.x86_64</td>
<td>shared-mime-info-0.70-6.el6.x86_64</td>
</tr>
<tr>
<td>smartmontools-5.43-1.el6.x86_64</td>
<td>snappy-1.1.0-1-6.el6.x86_64</td>
<td>sqlite-3.6.20-1-6.el6.x86_64</td>
</tr>
<tr>
<td>strace-4.5.19-1.19.el6.x86_64</td>
<td>sudo-1.8.6p3-15.el6.x86_64</td>
<td>sysstat-9.0.4-27.el6.x86_64</td>
</tr>
<tr>
<td>sysvinit-tools-2.87-5.dsfl.el6.x86_64</td>
<td>tar-1.23-11.el6.x86_64</td>
<td>tcp_wrappers-libs-7.6-57.el6.x86_64</td>
</tr>
<tr>
<td>tcpdump-4.0-0.3.20090921gtdf3cb4.2. el6.x86_64</td>
<td>telnet-0.17-48.el6.x86_64</td>
<td>tideway-10.2-505122.rhel6.dev.x86_64</td>
</tr>
<tr>
<td>tideway-appliance-6.15.05.61-503201. rhel6_64</td>
<td>tideway-device-api-4.2-1.rhel6.noarch</td>
<td>tideway-devices-4.2.2015.04.2-503996. ga.noarch</td>
</tr>
<tr>
<td>tideway-documentation-10.2-503455. rhel6.ga.noarch</td>
<td>traceroute-2.0.14-2.el6.x86_64</td>
<td>tw-atrium-1.2-1.rhel6.x86_64</td>
</tr>
<tr>
<td>tw-atss-8.1.00-1.5.rhel6.x86_64</td>
<td>tw-db-6.1.19-1.py2.7.rhel6.x86_64</td>
<td>tw-db-debuginfo-6.1.19-1.py2.7.rhel6. x86_64</td>
</tr>
<tr>
<td>tw-docutils-0.12-1.py2.7.rhel6.noarch</td>
<td>tw-egenix-mx-base-3.2.7-1.py2.7.rhel6. x86_64</td>
<td>tw-esmre-0.3.1-2.py2.7.rhel6.x86_64</td>
</tr>
<tr>
<td>tw-ipaddr-2.1.10-py2.7.rhel6.noarch</td>
<td>tw-jdk5-22-11.rhel6.x86_64</td>
<td>tw-jdk8-45-1.rhel6.x86_64</td>
</tr>
<tr>
<td>tw-jsvc-1.0.15-1.rhel6.x86_64</td>
<td>tw-M2Crypto-0.22.3-1.rhel6.x86_64</td>
<td>tw-omniORB-4.2.0.99.6278-5.py2.7.rhel6. x86_64</td>
</tr>
</tbody>
</table>

BMC Atrium Discovery 10.2  Page 116 of 1671
### Out of the box discovery capabilities

#### Applications

The number of applications that can be discovered by BMC Atrium Discovery increases with each monthly TKU release. This section contains information from Configipedia, which details the applications that can be discovered with the latest TKU installed. Clicking one of the links in this section will take you directly into Configipedia.

Tideway patterns can help you identify the configuration of runtime products in your data centers. Choose how you want to find the products of interest to you:

- **Product Name**
- **Publisher**
- **Product Category**
- **Business Applications**
Host attributes by platform

An overview of the attributes that BMC Atrium Discovery aims to recover from the different OS platforms is shown in the tables below. The attribute descriptions for each node kind are detailed in the relevant node sections later in this document.

The discovered attributes per OS platform for Inferred nodes is shown in the Table below. An X denotes that the attribute can be discovered. See OS platform key below.

The table shows the attributes from various OS platforms that can be discovered by a default BMC Atrium version 10.2 system. This does not include patterns or a TKU.

Some of my attributes are missing!

Some attributes might not be discovered on a given OS in your environment due to privilege requirements, hardware and software configurations, customizations, firewall configurations, security policies, and so forth. Additional attributes might be discovered by patterns, whether TKU provided, or your own custom patterns.

OS platform key

- A - AIX
- B - FreeBSD
- C - HPUX
- D - IBM i
- E - IRIX
- F - Linux
- G - MPE/iX
- H - Mac OS X
- I - NetBSD
- J - Netware
- K - OpenBSD
- L - OpenVMS
- M - POWER HMC
- N - Solaris
- P - Tru64
- R - UnixWare
- S - VMware ESX
- T - VMware ESXi
- U - Windows
| Node/Attribute          | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| **Host**               |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| cores_per_processor    | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |   |
| cpu_threading_enabled  | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| description            |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |
| dns_domain             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| domain                 | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| host_type              |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| hostid                 | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |
| hostname               | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| kernel                 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| ldom_name              |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| ldom_role              |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| logical_ram            | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| lpar_partition_number  | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| model                  | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| num_logical_processors| X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| num_processor_types    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| num_processors         | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os                     | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_arch                | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| os_build               |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_class               | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_edition             | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| os_level               | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| os_type                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_vendor              | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_version             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| processor_speed        | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| processor_type         | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| psu_status             |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| ram                    | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| serial                 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| Node/Attribute   | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| service_pack     |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| threads_per_core | X |   |   | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| uuid             |   |   |   |   | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |
| vendor           |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| workgroup        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| zone_uuid        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| zonename         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| Node/Attribute   | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| Patch            |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| name             | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X | X | X |
| Node/Attribute   | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| Package          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| arch             |   |   | X | X |   |   |   |   |   |   | X | X | X | X | X | X | X | X | X |
| description      |   |   |   |   | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| epoch            |   |   |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| name             |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| pkgname          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| revision         |   |   |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| vendor           |   |   | X | X |   |   |   |   |   |   |   | X | X | X | X |   |   |   |   | X |
| version          |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| Node/Attribute   | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| NetworkInterface |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| adapter_type     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| aggregated_by    |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| aggregated_with  |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| aggregates       |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| aggregation_mode |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| bonded           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| default_gateway  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| description      |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| dhcp_enabled     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| dhcp_server      |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| dns_servers      |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| Node/Attribute          | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| driver_date            |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |
| driver_version         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |
| duplex                 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| group_name             |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| ifindex                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| interface_type         | X |   | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| is_console             | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| mac_addr               | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| name                   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| negotiation            | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| physical_adapters      | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| physical_location      | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| ppa                    | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| primaryWins_server     | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| secondaryWins_server   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| service_name           | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| setting_id             | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| shared_adapters        | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| speed                  | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| virtual_adapters       | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| IPAddress              |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| broadcast              | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| fqdns                  | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| ip_addr                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| netmask                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| prefix                 | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| prefix_length          | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| temporary              | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| zone                   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| FibreChannelHBA        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

**Documentation**

BMC Software Confidential

BMC Atrium Discovery 10.2 Page 121 of 1671
### Node/Attribute

| Node/Attribute                  | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| aix_manufacturer_code         | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aix_part_number               | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| boot_bios                     |   | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| driver_name                   | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| driver_version                | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| firmware                      | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| hba_id                        | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| model                         |   | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| option_rom_version            | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| optrom_bios_version           | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| optrom_efi_version            | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| optrom_fcode_version          | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| optrom_fw_version             | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| serial                        |   | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

### FibreChannelNode

| Node/Attribute                  | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| firmware                      | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| wwnn                          | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |

### FibreChannelPort

| Node/Attribute                  | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| fabric_name                    |   | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| port_state                     | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| port_type                      | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| speed                          | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| supported_classes              | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| supported_speeds               | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |
| wwpn                           | X | X | X | X | X | X | X | X | X |   |   |   |   |   |   |   |   |   |   |

### SNMP devices

The number of SNMP devices that can be discovered by BMC Atrium Discovery increases with each monthly TKU release. This section contains information from Configipedia, which details the applications that can be discovered with the latest TKU installed. Clicking the links in this section will take you directly into Configipedia.
The following classes of SNMP device are supported.

- Network devices
- Printers
- SNMP managed devices

Monthly TKU releases might increase the SNMP device coverage.

Localization support

BMC Atrium Discovery supports the discovery of systems in any locale.

Report changes

New channels

The following new channels are provided.

Blade Enclosure Reports channel

- **All Blade Enclosures** – Shows all Blade Enclosures
- **Undiscovered Hosts for Processor Blades** – Shows Processor Blades without a related Host
- **Undiscovered Network Devices for I/O Modules** – Shows I/O Modules without a related Network Device. Note: some vendors, such as Dell, don't report I/O module information.

Management Controller Reports channel

- **Hosts with Management Controllers** – Shows all Hosts managed by a Management Controller
- **Management Controllers without Hosts** – Shows all Management Controllers which are not connected to a Host

Host Storage Reports channel

- **Host Storage** – Show Host Storage
- **SAN Storage Volumes used by Hosts** – Show SAN Storage Volumes in use by Hosts
- **Physical Disks used by Hosts** – Show Physical Disks in use by Hosts
- **Host Virtual Disks & Physical Storage** – Show Host Virtual Disks and the backing Physical Storage

Individual reports

- **Blade Enclosure Discovery Summary** – (Discovery Dashboard Reports) – Shows a summary of the discovery of each Blade Enclosure. Note that some vendors, such as Dell, do not report information on I/O Modules so the expected value can be zero.
• **Management Controllers near Removal Threshold** – (Discovery Reports - Model Health)
  – Show a list of Management Controller nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **Detailed CICS Transaction Programs** – (Infrastructure Reports - Mainframe Software) –
  Shows Detailed CICS Programs

**Context sensitive links**

The following new link is provided on Management Controller pages:

• **Subnets connected to device Device Name** — Subnets Connected to this Management Controller.

**Changes to third party software license terms**

BMC Atrium Discovery uses some open source and freely distributable binary components. The terms of their licenses are available from Customer Support.

**Changes to license terms in BMC Atrium Discovery version 10.2**

The following products have been upgraded in BMC Atrium Discovery Version 10.2 and their licenses have been updated. For information, contact Customer Support.

<table>
<thead>
<tr>
<th>Application</th>
<th>License type</th>
<th>10.1</th>
<th>10.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>docutils</td>
<td>docutils</td>
<td>0.11</td>
<td>0.12</td>
</tr>
<tr>
<td>M2Crypto</td>
<td>BSD</td>
<td>0.21.1</td>
<td>0.22.3</td>
</tr>
<tr>
<td>python</td>
<td>Python Software Foundation</td>
<td>2.7.8+patches</td>
<td>2.7.9+patches</td>
</tr>
<tr>
<td>python-ldap</td>
<td>python-ldap</td>
<td>2.4.13</td>
<td>2.4.19</td>
</tr>
<tr>
<td>postgresql</td>
<td>PostgreSQL</td>
<td>9.2-1002.jdbc4</td>
<td>9.4-1200.jdbc4</td>
</tr>
<tr>
<td>jtds SQL and Sybase</td>
<td>LGPL</td>
<td>1.2.7</td>
<td>1.3.1</td>
</tr>
<tr>
<td>Berkeley DB</td>
<td>Berkeley DB License (commercial)</td>
<td>6.0.20</td>
<td>6.1.19</td>
</tr>
<tr>
<td>tomcat</td>
<td>Apache v2.0</td>
<td>6.0.41</td>
<td>6.0.43</td>
</tr>
<tr>
<td>openssl</td>
<td>OpenSSL</td>
<td>1.0.1j</td>
<td>1.0.1l</td>
</tr>
<tr>
<td>openssl-fips</td>
<td>OpenSSL</td>
<td>2.0.8</td>
<td>2.0.9</td>
</tr>
<tr>
<td>d3.js</td>
<td>MIT</td>
<td>3.4.11</td>
<td>3.5.5</td>
</tr>
<tr>
<td>cola.js</td>
<td>MIT</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>NMAP</td>
<td>GPL</td>
<td>6.40</td>
<td>6.47</td>
</tr>
</tbody>
</table>

**Changes to license terms in BMC Atrium Discovery version 10.1**

The following products have been upgraded in BMC Atrium Discovery Version 10.0 and their licenses have been updated. For information, contact Customer Support.
The omniORB GPL/LGPL license indicates that the stub code produced by the IDL compiler is not considered a derived work of it.

The NMAP (GPL license) was downgraded back to version 6.40.

Changes to license terms in BMC Atrium Discovery version 10.0

The following products have been upgraded in BMC Atrium Discovery Version 10.0 and their licenses have been updated. For information, contact Customer Support.

- bsddb3: upgraded from 5.3.0 to 6.0.1
- Berkeley DB: upgraded from 5.3.15 to 6.0.20
- docutils: upgraded from 0.8.1 to 0.11
- egenix-mx-base: upgraded from 3.2.4 to 3.2.7
- innosetup: upgraded from 5.5.3 to 5.5.4
- iozone3: upgraded from 3-398 to 3-420
- jsvc: upgraded from 1.0.10 to 1.0.15
- JDK 6: removed
- JDK 7: upgraded from 1.7.0_25 to 1.7.0_51
- NMAP: upgraded from 6.25 to 6.40
- omniORB: upgraded from 4.1.99.6105 to 4.1.99.6243
- omniORBpy: upgraded from 4.1.99.6105 to 4.1.99.6243
- openssl: upgraded from openssl-1.0.0-27 to 1.0.1f
- python: upgraded from 2.7.3 to 2.7.6
- python-ldap: upgraded from 2.3.13 to 2.4.13
- reportlab: upgraded from 2.5 to 2.7
- tomcat: upgraded from 6.0.36 to 6.0.37
- tripwire: upgraded from 2.4.2 to 2.4.2.2
• Webware: upgraded from 1.1 to 1.1.1

Windows discovery utilities no longer shipped in BMC Atrium Discovery

From version 8.1 the following utilities are no longer shipped with BMC Atrium Discovery. You can download them freely and install them where required.

• rcmd: see the Microsoft Download Center and search for the resource kit for the version of Windows that you are running.
• pulist: download this from the Microsoft Download Center.
• tlist: download this from the Microsoft Download Center.

BMC Atrium Discovery copyright information

 Portions of the software installed are:

Apache

Copyright © 1999-2015 The Apache Software Foundation

Apache License, Version 2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or
contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and

2. You must cause any modified files to carry prominent notices stating that You changed the files; and

3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
into this Agreement ("Customer") and the BMC Entity for the applicable Region where Customer acquired the License as described in Section 18 ("BMC").

1. GENERAL DEFINITIONS.
"Affiliate" is an entity that controls, is controlled by or shares common control with BMC or Customer, where such control arises from either (a) a direct or indirect ownership interest of more than 50% or (b) the power to direct or cause the direction of the management and policies, whether through the ownership of voting stock by contract, or otherwise, equal to that provided by a direct or indirect ownership of more than 50%.
"Documentation" means the technical publications relating to the software, such as release notes, reference, user, installation, systems administrator and technical guidelines, included with the Product.
"Licensed Capacity" is the amount of each Product licensed as established in the Order.
"Order" is an agreed written or electronic document, subject to the terms of this Agreement that identifies the Products to be licensed and the Licensed Capacity and/or the Support to be purchased and the fees to be paid.
"Product" is the object code of the software and all accompanying Documentation delivered to Customer, including all items delivered by BMC to Customer under Support.
"Support" is the support services program as further specified in this Agreement.
"Territory" means the country(ies) where Customer is licensed to install the Product as specified in the Order.

2. SCOPE. Licenses are granted, and Support is obtained, solely by execution of Orders. Each Order is deemed to be a discrete contract, separate from each other Order, unless expressly stated otherwise therein, and in the event of a direct conflict between any Order and the terms of this Agreement, the terms of the Order will control only if the Order is agreed to by each party. Orders may be entered under this Agreement by and between (a) BMC or an Affiliate of BMC; and (b) the Customer or an Affiliate of Customer. With respect to an Order, the terms "BMC" and "Customer" as used in this Agreement will be deemed to refer to the entities that execute the Order, the Order will be considered a two party agreement between such entities, and BMC or its authorized reseller will separately invoice the Customer named in the Order for the associated License fees and Support fees. Neither execution of this Agreement, nor anything contained herein, shall obligate either party to enter into any Orders. In the event an Order is proposed by BMC and is deemed to constitute an offer, then acceptance of such offer is limited to its terms. In the event Customer proposes an Order by submitting a purchase order, then regardless of whether BMC acknowledges, accepts or fully or partially performs under such purchase order, BMC OBJECTS to any additional or different terms in the purchase order.

3. LICENSE. Subject to the terms, conditions, payment requirements and restrictions set forth in this Agreement, BMC grants Customer a non-exclusive, non-transferable, non-sub-licensable perpetual (unless a non-perpetual license is provided on an Order) license to install in the Territory, access and use the Product (i) up to the Licensed Capacity,
(ii) for Customer's and its Affiliates internal business operations, (iii) in accordance with the Documentation and the applicable Order, and (iv) make one copy of the Product for archival purposes only (collectively a "License"). Affiliates may use and access the Products and Support under the terms of this Agreement, and Customer is responsible for its Affiliates compliance with the terms of this Agreement.

4. RESTRICTIONS. Customer will not: (a) copy, operate or use any Product in excess of the applicable Licensed Capacity or other than as set forth in the License above; (b) modify, delete or remove any ownership, title, trademark, patent or copyright notices from any Product, or copy or partial copy of a Product; (c) disassemble, reverse engineer, decompile or otherwise attempt to derive any Product source code from object code, except to the extent expressly permitted by applicable law despite this limitation without possibility of contractual waiver; (d) distribute, rent, lease, sublicense or provide the Product to any third party; (e) use the Products in an outsourcing or service bureau environment on behalf of non-Affiliate third parties, or allow the products to be used by an outsourcing or service bureau provider on Customer's behalf; (f) provide a third party with the results of any functional evaluation, or performance tests, without BMC's prior written approval; or (g) attempt to disable or circumvent any of the licensing mechanisms within the Product.

5. PRODUCT PERFORMANCE WARRANTY. BMC warrants that (a) the Product will perform in substantial accordance with its Documentation for a period of one year from the date of the first Order, (b) BMC has used commercially reasonable efforts consistent with industry standards to scan for and remove software viruses, and (c) other than passwords that may be required for the operation of the Product, BMC has not inserted any code that is not addressed in the Documentation and that is designed to delete, interfere with or disable the normal operation the Products in accordance with the License. This warranty will not apply to any problems caused by hardware, Computers, or software other than the Product, or misuse of the Product, use of the Product other than as provided by the applicable License, modification of the Product, or claims made either outside the warranty period or not in compliance with the notice and access requirements set forth below. No warranty is provided for additional Licensed Capacity, Product provided pursuant to Support or Product provided pursuant to Section 12.

6. LIMITED REMEDIES. BMC's entire liability, and Customer's exclusive remedy, for breach of the above warranty is limited to: BMC's use of commercially reasonable efforts to have the Product perform in substantial accordance with its Documentation, or replacement of the non-conforming Product within a reasonable period of time, or if BMC cannot have the Product perform in substantial accordance with its Documentation replace the Product within such time period, then BMC will refund the amount paid by Customer for the License for that Product. Customer's rights and BMC's obligations in this Section are conditioned upon Customer's providing BMC during the warranty period (a) full cooperation and access to the Product in resolving any claim; and (b) written notice addressed to the BMC Legal Department that includes notice
of the claim, a complete description of the alleged defects sufficient to permit their reproduction in BMC's development or support environment, and a specific reference to the Documentation to which such alleged defects are contrary.

7. DISCLAIMER OF WARRANTIES. EXCEPT FOR THE EXPRESS WARRANTIES IN THIS AGREEMENT, THE PRODUCT IS PROVIDED WITH NO OTHER WARRANTIES WHATSOEVER, AND BMC, ITS AFFILIATES AND LICENSORS DISCLAIM ALL OTHER WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. BMC DOES NOT WARRANT THAT THE OPERATION OF THE PRODUCT WILL BE UNINTERRUPTED OR ERROR FREE, OR THAT ALL DEFECTS CAN BE CORRECTED.

8. SUPPORT. Customer may acquire BMC support services ("Support") on an Order for the total Licensed Capacity of a Product. Once Support is acquired for a Product, Customer is automatically enrolled in Support on an annual basis for all Licensed Capacity of that Product, unless either party terminates Support on all Licensed Capacity of a Product upon at least 30 days written notice prior to the next Support anniversary date. The annual fee for Support will be agreed upon at the time of each Order. A further description of Support is located at www.bmc.com/support/review-policies, and is incorporated herein by this reference. BMC may change its Support terms to be effective upon Customer's support anniversary date. BMC reserves the right to discontinue Support for a Product where BMC generally discontinues such services to all licensees of that Product. If Customer terminates Support and then re-enrolls in Support, BMC may charge Customer a reinstatement fee.

9. PAYMENT, DELIVERY AND TAXES. If Customer is purchasing directly from BMC, Customer will pay each License fee and/or Support fee upon receipt of invoice. Customer will pay, or reimburse, BMC or when required by law the appropriate governmental agency for taxes of any kind, including sales, use, VAT, excise, customs duties, withholding, property, and other similar taxes (other than taxes based on BMC's net income) imposed in connection with the License and/or the Support fees which are exclusive of these taxes. For Products that are delivered electronically, upon request from BMC, Customer agrees to provide BMC with documentation supporting that the designated Product was received electronically. If Customer accepts any Product in a non-electronic format, there may be an additional charge and it is the sole responsibility of Customer to bear any sales/use tax obligation, penalties, and interest. The unpaid balance of each late payment bears interest at a rate equal to the lesser of 1% per month or the maximum amount permitted by law. All Products are licensed FCA ("Free Carrier" as per Incoterms 2000) shipping point. The Products are accepted on the date BMC delivers the Product to the Customer either physically or by providing access codes for electronic download, whichever occurs first, however, such acceptance will not affect the Product Performance Warranty provided in this Agreement.

10. PROPRIETARY RIGHTS AND CONFIDENTIALITY. (a) BMC, its Affiliates or licensors retain all right, title and interest to the Product, Support and all related intellectual property and proprietary rights. The Product and all third party software provided with the Product are protected by
applicable copyright, trade secret, industrial and other intellectual property laws. BMC reserves any rights not expressly granted to Customer in this Agreement. (b) "Confidential Information" means all proprietary or confidential information that is disclosed to the recipient ("Recipient") by the discloser ("Discloser"), and includes, among other things (i) any and all information relating Discloser financial information, customers, employees, products or services, including, without limitation, software code, flow charts, techniques, specifications, development and marketing plans, strategies, forecasts, and proposal related documents and responses; (ii) as to BMC, and its licensors, the Product (excluding portions of the Documentation that BMC makes publicly available) and any third party software provided with the Product; and (iii) the terms of this Agreement, including without limitation, Product pricing information. Confidential Information does not include information that Recipient can show: (a) was rightfully in Recipient's possession without any obligation of confidentiality before receipt from the Discloser; (b) is or becomes a matter of public knowledge through no fault of Recipient; (c) is rightfully received by Recipient from a third party without violation of a duty of confidentiality; or (d) is independently developed by or for Recipient. Recipient may not disclose Confidential Information of Discloser to any third party or use the Confidential Information in violation of this Agreement. The Recipient (i) will exercise the same degree of care and protection with respect to the Confidential Information of the Discloser that it exercises with respect to its own Confidential Information and (ii) will not, either directly or indirectly, disclose, copy, distribute, republish, or allow any third party to have access to any Confidential Information of the Discloser. Notwithstanding the foregoing, Recipient may disclose Discloser's Confidential Information to Recipient's employees and agents who have the need to know provided that such employees and agents have legal obligations of confidentiality substantially the same (and in no case less protective) as the provisions of this Agreement. (c) Notification Obligation. If the Recipient becomes aware of any unauthorized use or disclosure of Discloser's Confidential Information, then Recipient will promptly and fully notify the Discloser of all facts known to it concerning such unauthorized use or disclosure. In addition, if the Recipient or any of its employees or agents are required (by oral questions, interrogatories, requests for information, or documents in legal proceedings, subpoena, civil investigative demand, or other similar process) to disclose any of Discloser's Confidential Information, the Recipient will not disclose the Discloser's Confidential Information without providing the Discloser with commercially reasonable advance prior written notice to allow Discloser to seek a protective order or other appropriate remedy or to waive compliance with this provision. In any event, the Recipient will exercise its commercially reasonable efforts to preserve the confidentiality of the Discloser's Confidential Information, including, without limitation, cooperating with Discloser to obtain an appropriate protective order or other reliable assurance that confidential treatment will be accorded to the Confidential Information.

11. DISCLAIMER OF DAMAGES; LIMITS ON LIABILITY. EXCEPT FOR VIOLATIONS OF LICENSE RESTRICTIONS (SECTION 4), AND PROPRIETARY RIGHTS AND CONFIDENTIALITY (SECTION 10) AND FOR INFRINGEMENT CLAIMS (SECTION 13),
NEITHER PARTY, ITS AFFILIATES OR BMC'S LICENSORS ARE LIABLE FOR (A) ANY SPECIAL, INDIRECT, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES RELATING TO OR ARISING OUT OF THIS AGREEMENT, SUPPORT, THE PRODUCT OR ANY THIRD PARTY CODE OR SOFTWARE PROVIDED WITH THE PRODUCT (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, LOST COMPUTER USAGE TIME, AND DAMAGE TO, OR LOSS OF USE OF DATA), EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, AND IRRESPECTIVE OF NEGLIGENCE OF A PARTY OR WHETHER SUCH DAMAGES RESULT FROM A CLAIM ARISING UNDER TORT OR CONTRACT LAW OR (B) DAMAGES OF ANY KIND IN AN AMOUNT GREATER THAN THE AMOUNT OF ACTUAL, DIRECT DAMAGES UP TO THE GREATER OF THE AMOUNT PAID AND PAYABLE BY CUSTOMER FOR THE LICENSE TO THE APPLICABLE PRODUCT GIVING RISE TO SUCH DAMAGES.

12. TRIAL LICENSE. BMC may determine, in its sole discretion, to make products available to Customer without an Order and without charge. Such products are deemed to be "Products" pursuant to this Agreement except that (a) they are provided to Customer solely so that Customer may evaluate internally whether to acquire a license to the products for a fee, (b) the license term for such products is 30 days; (c) the Products are provided "AS IS" and without any warranty or support, and (d) the products cannot be put into productive use or included as part of Customer's business processes in any manner, unless or until they are expressly licensed and paid for under an Order. BMC may terminate all of Customer's rights and licenses to these Products for BMC's convenience upon notice to Customer.

13. INFRINGEMENT CLAIMS. If a third party asserts a claim against Customer asserting that Customer's use of a Product in accordance with this Agreement violates that third-party's patent, trade secret or copyright rights ("Infringement Claim"), then BMC will, at its own expense: (a) defend or settle the Infringement Claim; and (b) indemnify Customer for any damages finally awarded against Customer based on infringement by the Product. BMC's obligations under this Section will not apply if: (a) BMC's legal department does not receive prompt, detailed written notice of the Infringement Claim from Customer, (b) BMC is not able to retain sole control of the defense of the Infringement Claim and all negotiations for its settlement or compromise, (c) BMC does not receive all reasonable assistance, or (d) the Infringement Claim is based on (i) the use of Product in combination with products not approved by BMC in the Product's Documentation, (ii) the failure of Customer to use any updates to such Product within a reasonable time after such updates are made available to Customer, or (iii) the failure of Customer to use the Product as permitted by the Order and in accordance with the Documentation. BMC will not bind Customer to a monetary obligation in a settlement or compromise, or make an admission on behalf of Customer, without obtaining Customer's prior consent. If BMC determines in BMC's reasonable discretion that use of the Product should be stopped because of an Infringement Claim or potential Infringement Claim, if a court of competent jurisdiction enjoins Customer from using a Product as a result of an Infringement Claim and BMC is unable to have such injunction stayed or overturned, or if BMC settles an Infringement Claim on terms that would require Customer to stop using the Product, then BMC will, at its expense and election: (a) modify or replace the Product, (b) procure the right to continue using the Product, or (c) if in BMC's reasonable judgment, neither (a) or (b) is commercially reasonable, terminate
Customer's License to the Product and (i) for any perpetual licenses, issue a refund based upon the applicable license fees paid, prorated over 48 months from the date of the Order under which the Products were initially licensed; and (ii) for any non-perpetual licenses, release Customer from its obligation to make future payments for the Product or issue a pro rata refund for any fees paid in advance. This Section contains Customer's exclusive remedies and BMC's sole liability for Infringement Claims.

14. TERMINATION. Upon thirty days advance written notice, either party may terminate this Agreement for its convenience on a prospective basis; however, such termination will have no effect on Orders placed prior to its effective date and such Orders will remain in full force and effect under the terms of this Agreement. BMC may: (i) terminate an Order and the Licenses to the Products on that Order if Customer fails to pay any applicable fees due under that Order within 30 days after receipt of written notice from BMC of non-payment; (ii) terminate any or all Orders, Licenses to the Products and/or this Agreement, without notice or cure period, if Customer violates the intellectual property rights of BMC, its Affiliates or licensors, or uses the Products outside of the scope of the applicable Licenses; or (iii) terminate all Licenses and this Agreement in whole or in part if Customer commits any other material breach of this Agreement and fails to correct the breach within 30 days after BMC notifies Customer in writing of the breach. Upon any termination of a License, Customer will immediately uninstall and stop using the relevant Product, and upon BMC's request, Customer will immediately return such Product to BMC, together with all related Documentation and copies, or certify its destruction in writing.

15. AUDIT. If requested by BMC, and not more than once a year, Customer agrees to deliver to BMC, within 30 days of such request, as specified by BMC either (a) periodic product usage reports generated from specific products or (b) written periodic product usage reports, to be provided solely when the product does not generate reports. Additionally, if requested by BMC not more than once a year, Customer agrees to allow BMC to perform an audit at the locations where the Products are installed, during normal business hours to ensure compliance with the terms of this Agreement. Customer agrees to cooperate during any such audit and to provide reasonable access to its information and systems. If an audit reveals that Customer has exceeded the Licensed Capacity for a Product, Customer agrees to pay the applicable fees for additional capacity upon receipt of invoice. If the understated capacity exceeds 5% of the Licensed Capacity of the applicable Product, then Customer agrees to also pay BMC's reasonable costs of conducting the audit.

16. EXPORT CONTROLS. Customer represents and warrants that it: a) will comply with the United States Export Administration Regulations and other U.S. or foreign export regulations; b) no individual accessing or using the Product is a citizen of or from an embargoed country (currently Iran, Syria, Sudan, Cuba and North Korea); c) is not prohibited from receiving the Product under such regulations; d) will not acquire the Product for a person who is restricted under such regulations; e) will not use the Product in contradiction to such regulations; and f) will not use the Product for prohibited uses, including but not limited to nuclear,
chemical, missile or biological weapons related end uses. For Product exported from Ireland, EC No. 428/2009 sets up a Community regime for control of exports of dual-use items and technology, and it is declared that this Product is intended for civil purposes only. Therefore, Customer agrees to comply with both the U.S. regulations and those E.U. regulations and will not export in violation of the regulations and without all proper licenses. Any failure to comply with these regulations will result in Customer forfeiting all rights to the Product.

17. GOVERNING LAW AND DISPUTE RESOLUTION. A party will provide written notice to the other party of any controversy, dispute or claim arising out of or relating to this Agreement, or to the formation, interpretation, breach, termination, or validity thereof ("Controversy"). The parties shall engage in good faith negotiations to resolve the Controversy. Only if the Controversy is not resolved through good faith negotiations within 15 days of the sending of the written notice of Controversy, the Controversy may be submitted to litigation or binding arbitration, based on the place of incorporation of the parties, as follows:

(i) If both parties to this Agreement are entities incorporated under the law of any state in the United States, the Controversy shall be tried in either state or federal court located in Houston, Texas and the laws of the State of Texas shall govern. Both sides hereby submit to the exclusive jurisdiction of the courts in Houston, Texas and waive all defenses based on forum non conveniens.
(ii) If both parties to this Agreement are entities incorporated in countries in the Europe, Middle East, or Africa regions, the arbitration shall be held in Amsterdam, Netherlands under the then-applicable rules of the International Chamber of Commerce and the substantive laws of the Netherlands will govern.
(iii) If both parties to this Agreement are entities incorporated in countries in the Asia Pacific region, the arbitration shall be held in Singapore under the then-applicable rules of the Singapore International Arbitration Centre and the substantive laws of Singapore will govern.
(iv) In all other instances, the arbitration shall be held in New York City, New York, under the then-applicable international rules of the American Arbitration Association and the substantive laws of the State of Texas will govern.

For all arbitrations conducted hereunder: (a) the arbitration shall be conducted in English; (b) the relevant arbitral institution shall determine the number of arbitrators, but any Controversy in which the amount in dispute is greater than $10 million USD shall be decided by three arbitrators, with each party having the right to select one arbitrator; (c) the costs of such arbitration shall be borne equally, pending the arbitrator's award; (d) the arbitration award rendered shall be final and binding on the parties, shall not be subject to appeal to any court and shall be enforceable in any court having jurisdiction over the Parties; (e) the arbitration proceedings, award and pleadings shall all be confidential, unless disclosure of particular information is required for purposes of enforcing/challenging the award or to meet local securities law requirements; and (f) the party prevailing in arbitration shall be entitled to recover its reasonable attorneys' fees and the necessary costs incurred in connection with the arbitration.

The United Nations Convention on Contracts for the International Sale of
Goods shall not apply to this Agreement. Nothing in this Agreement shall be deemed as preventing either party from seeking immediate injunctive relief from any court having jurisdiction over the parties and the subject matter of the dispute.

18. BMC ENTITIES. The following licensing entities apply to this Agreement:

<table>
<thead>
<tr>
<th>Region Licensing Entity</th>
<th>Address of Licensing Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>BMC Software, Inc. 2101 CityWest Boulevard, Houston, Texas 77042</td>
</tr>
<tr>
<td>South (not a specified Central or South America country below)</td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td>BMC Software Canada Inc. 50 Minthorn Boulevard, Suite 200 Markham, Ontario L3T 7X8 Canada</td>
</tr>
<tr>
<td>EMEA (Europe, Middle East and African Countries)</td>
<td>Middle East and Distribution B.V. 1119 PE Schiphol Rijk, The Netherlands</td>
</tr>
<tr>
<td>Brazil</td>
<td>BMC Software do Rua Leopoldo Couto de Brasil Ltda. Magalhães Jr, 758 - 14º andar, São Paulo - SP - Brazil</td>
</tr>
<tr>
<td>Mexico</td>
<td>BMC Software Distribution Torre Esmeralda II Blvd. de México, S.A. de C.V. Manuel Avila, Camacho #36, Piso 23 Lomas de Chapultepec, CP11000, México D.F.</td>
</tr>
<tr>
<td>Argentina</td>
<td>BMC Software de Ing. Butty 220 - Piso 14, Argentina S.A. Buenos Aires, República Argentina, C1001AFB</td>
</tr>
<tr>
<td>Southeast Asia (S.E.A)</td>
<td>S.E.A BMC Software Asia Pacific 600 North Bridge Road, Suite 501-504, Level 5, Limited Tower W1, The Towers, Oriental Plaza, #1 East Chang An Ave., Dong Cheng, Beijing 100738, China</td>
</tr>
<tr>
<td>Hong Kong, Taiwan</td>
<td>Japan BMC Software K.K. Harmony Tower 24th Floor,</td>
</tr>
</tbody>
</table>
19. ASSIGNMENT AND TRANSFERS. Customer may not assign or transfer a Product separate from the applicable Agreement and License, and may not assign or transfer an Agreement or a License, except in the event of a merger with or into, or a transfer of all or substantially all of Customer's assets to, a third party who assumes all of Customer's liabilities and obligations under the Agreement and License, and expressly agrees in writing to be bound by and comply with all of the terms of the Agreement and License. Except as specifically authorized by applicable law, any attempt to assign or transfer an Agreement or License in violation of this provision will be null and void and be treated as a violation of BMC's intellectual property rights or use outside the scope of the License.

20. DATA PROTECTION. (a) Customer acknowledges that BMC neither requires nor needs Customer to (i) send BMC any personal data collected by Customer ("Customer Collected Data") or (ii) give BMC access to any Customer Collected Data. Consequently, Customer remains responsible for either filtering, making anonymous, encrypting such Customer Collected Data or for having proper procedures in place to prevent Customer Collected Data from being sent to or accessed by BMC. (b) In the course of normal business, BMC may collect and process personal information related to the Customer (mainly contact and related information) in order to perform its obligations under this Agreement and/or under an Order, such information being referred to hereinafter as "Customer Contact Information". BMC undertakes to comply with all the relevant data protection legislation and/or regulations where the Customer Contact Information is to be processed. BMC shall in particular: (i) allow Customer to access, modify, correct or erase Customer Contact Information when necessary; (ii) take reasonable technical and organizational security measures to maintain the confidentiality and integrity of Customer Contact Information and to prevent its unauthorized access, use, or disclosure; (iii) refrain from transferring any Customer Contact Information to any third party without obtaining the express consent of the person to whom Customer Contact Information relates to, and (iv) refrain from using Customer Contact Information for any other purpose than performing its obligations under this Agreement and/or any Order.

21. MISCELLANEOUS TERMS. A waiver by a party of any breach of any term of this Agreement will not be construed as a waiver of any continuing or succeeding breach. Should any term of this Agreement be invalid or unenforceable, the remaining terms will remain in effect. The parties acknowledge they have read this Agreement and agree that it is the complete and exclusive statement of the agreement and supersedes any prior or contemporaneous negotiations or agreements, between the parties relating to the subject matter of this Agreement. There are no
representations, promises, warranties, covenants, or undertakings between the parties other than those expressly set forth in this Agreement. This Agreement may not be modified or rescinded except in writing signed by both parties. Any delay or failure of any party to perform any obligation under this Agreement caused by governmental restrictions, labor disputes, storms or natural disasters, emergency, or other causes beyond the reasonable control of the party, will not be deemed a breach of this Agreement; provided, however, this provision does not apply to the payment of monies or any breach of Section 10. Customer agrees that BMC and its affiliates may refer to Customer as a customer of BMC, both internally and in externally published media. The BMC Products may contain third party software which is delivered to Customer as part of the Product and may not be taken out of the Product or used separately from the Product and for which additional terms may be included in the Documentation. Any additional documents presented to a BMC representative by Customer for signature as a condition for going on a Customer's site will be governed by this Agreement and to the extent that such document presents additional terms or conflicts with this Agreement, it shall be considered null and void.

22. U.S. FEDERAL ACQUISITIONS. This Section applies only to acquisitions of the commercial Product and Documentation subject to this Agreement by or on behalf of the United States Government, or by any prime contractor or subcontractor (at any tier) under any contract, grant, cooperative agreement or other activity with the United States Government. In the event the Products are delivered to the United States Government, the United States Government hereby agrees that the Products qualify as "commercial items" within the meaning of the Federal acquisition regulation(s) applicable to this procurement. The terms and conditions of this Agreement shall pertain to the United States Government's use and disclosure of the Product, and shall supersede any conflicting contractual terms and conditions. The following additional statement applies only to acquisitions governed by DFARS Subpart 227.4 (October 1988): "Restricted Rights - Use, duplication and disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 (Oct. 1988)."

23. ADDITIONAL TERMS. The following additional terms are incorporated into this Agreement.

a. DEFINITIONS. Terms set forth below have the indicated meaning regardless of whether they are capitalized.
"Client" means a third party whose data is processed by Customer and is only permitted if Customer is an authorized BMC service provider.
"Cloud Environment" means a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) managed so they behave as if they were one computer.
"Cloud Service Provider" is an entity that provides Cloud Services to Clients under agreements pursuant to transactions for which the Cloud Service Provider is compensated.
"Cloud Services" means the dynamic provisioning of IT resources as a service, where typically the Cloud infrastructure is shared across multiple tenants, and tenants are billed on a utility/subscription basis
for what they use. Examples of Cloud Services include Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

"Computer" or "Server" has the meaning generally given within the computer industry, which is a single machine, whether a central processing unit, such as a mainframe machine, or a distributed systems machine, such as a Unix or Intel based server. A mainframe machine would be an individual mainframe computer having single or multiple processors or engines. For purposes of distributed systems machines a Computer or Server may be physical or virtual.

"Documentation" means the technical publications relating to the software, such as release notes, reference, user, installation, systems administrator and technical guidelines, included with the Product.

"Enterprise" is the environment consisting of all hardware owned or leased by a Customer or by a Client respectively, in the Territory.

b. LICENSE RESTRICTIONS. The following restrictions apply to certain Products.


AppSight Products:
* BMC AppSight Initial Platform ("AppSight System") may only be used to support a Customer’s own applications according to the AppSight System configuration licensed. For this purpose, Customer’s "own" applications are those of which Customer is the ultimate beneficiary or applications developed by Customer as an independent software vendor. Each AppSight System may only be used with the platform designated on the Product Table (Windows/.Net or J2EE) unless a BMC AppSight Additional Platform is licensed for that AppSight System which enables the AppSight System to be used with an additional platform. Each AppSight System may only be used for the Workflow designated on the Product Order Form unless a BMC AppSight Additional Platform is licensed for that AppSight System for an additional Workflow.
  o AppSight System-Team Edition may only be used at a designated Site to support a designated Application for a designated Workflow.
  o AppSight System - Group Edition may be used at a designated Site to support multiple applications of a designated Group for a designated Workflow.
  o AppSight System - Division Edition may be used at multiple Sites to support multiple applications of a designated Division for a designated Workflow.
* BMC AppSight Named User is a license to use the full BMC AppSight System. Customer must provide the individual domain ID or email ID for each named user. Once designated, the BMC AppSight Named User may only be changed if the individual leaves the Customer or the Team, Group or Division as applicable. BMC AppSight Named Users may not be transferred from one AppSight System to another.
* BMC AppSight QA User is a license to use the AppSight Test Recorder module of the AppSight System.
* BMC AppSight Level 1 Viewer is a license to use limited-functions of AppSight System that provides a user with the ability to view and replay the visual recording of a Black Box log. The AppSight Level 1 Viewer may be used by the named users provided to BMC.
* BMC AppSight for Citrix Support User is a license to use the AppSight
System for Citrix application support. Only licensed BMC AppSight for Citrix users may use the AppSight System for Citrix support.

* BMC AppSight Connector for Defect/Incident Tracking may only be used to interface the AppSight System to Customer's defect/incident tracking application.

* BMC Desktop Capture. Customer must provide the individual domain ID or email ID for each named user. The BMC Desktop Capture Player may be used by licensed named users to play back and view BMC Desktop Capture incident recordings. Once designated, a named user may only be changed if the individual leaves the Customer.

BMC AppSight and BMC Application Problem Resolution Products:

* License Key. The Product may require a software license key that limits usage to that provided under the terms of the Agreement and this Product Order Form. Licensee must install and run license manager software provided with the Product at no additional cost. All AppSight Consoles must be connected to the license manager.

* AppSight Black Box is the agent portion of the AppSight System. All AppSight Systems include unlimited AppSight Black Boxes which may be installed on any computers of Customer or Customer's customers. Customer may provide Customer's customers the limited right to install and to use on behalf of Customer the AppSight Black Boxes (but not any other components of the AppSight System) but only for the sole purpose of providing information to Customer to support Customer's own software applications, and in no event to support Customer's customer's own internal software applications.

* Embedded Black Box. Customer may incorporate the AppSight Black Box into a Customer application, in which case, in addition to the licenses granted above, Customer is hereby granted a worldwide, nonexclusive, perpetual license to (a) incorporate the unmodified unaltered object code version of the AppSight Black Box into Customer's designated application ("Customer Application"); (b) reproduce and distribute the AppSight Black Box as incorporated into the Customer Application; and (c) to use in unaltered form the BMC trademarks, service marks or marketing logos (the "BMC Trademarks") solely to promote the Customer Application, provided Customer obtains BMC's prior written approval for each new usage. Customer shall ensure that any Customer Application incorporating the AppSight Black Box shall be governed by a license agreement which is at least as protective of BMC's proprietary rights in the AppSight Black Box as of Customer's proprietary rights in the Customer Application, but no less protective than this Agreement, including rights and restrictions related to end user's right to make backup and archival copies. In the event Customer incorporates the Black Box into Customer's Application, Customer shall include in the startup screen, help and/or the about screens in the Customer Application, BMC's logo and the following, "POWERED BY BMC'S APPSIGHT BLACK BOX TECHNOLOGY. " Furthermore, Customer shall visually display the BMC name and the BMC product names and trademarks in the documentation for the Customer Application incorporating the AppSight Black Box, on Customer's website and in advertising and promotional materials.

BMC Atrium Orchestrator - Adapters License Add-on: Restricts license rights to deploy one unique adapter for every unit licensed; also includes unlimited deployment rights to use Light Weight Activity Peers in combination with licensed adapters; test and development license are
* Adapter - a system/interfaces/gateways/connectors used to talk to external applications.

* Light Weight Activity Peer - Slave peers/servers that can optionally be added to a grid to accommodate network latencies and/or security topologies when deployed in combination with adapters; these peers do not directly add incremental processing power.

BMC Atrium Orchestrator Automation Pack - Device Endpoint License Add-on:
Restricts license rights to up to five (5) peers in the orchestration environment; and allows use of the following runbooks (Continuous Compliance for Network Automation and NetApp Solutions) and supporting base Adapters only for the Licensed Capacity.

BMC Atrium Orchestrator Automation Pack - Server Endpoint License Add-on:
Restricts license rights to up to five (5) peers in the orchestration environment; and allows use of the following runbooks (Continuous Compliance for Server Automation, Discovery Synch, and NetApp Solutions) and supporting base Adapters only for the Licensed Capacity.

BMC Atrium Orchestrator - Peer License Add-on: Restricts license rights to deploy one peer for every unit licensed; a peer can either be a Configuration Distribution Peer or an Application Peer; test and development license are provided at no additional cost.

* Application Peer - Server that executes workflows.

* Configuration Distribution Peer - Master application that controls all workflows, including load balancing across the grid of Peers.


BMC BladeLogic Automation Suite - Base License, BMC Remedy IT Service Management, BMC ProactiveNet Performance Management Suite - Base License or BMC TrueSight Operations Management - Base License: Includes unlimited use of the following BMC Atrium Orchestrator core product components including Repository, Access Manager, Grid Manager, Metrics and Reporting, Operations Actions Management Modules, and Operations Actions Utility Modules. Test and development licenses are provided at no additional cost.

Restricts license rights to use the following BMC Atrium Orchestrator product components for configuration of an initial Orchestration environment as detailed here: One (1) Configuration Distribution Peer, 3 Development Studio user licenses, 5 Operator Control Panel user licenses, and Application Adapters for BMC products.

* Adapter - a system/interfaces/gateways/connectors used to talk to external applications

* Configuration Distribution Peer - Master application that controls all workflows, including load balancing across the grid of Peers.

* Development Studio - Graphical workflow authoring tool used to create and modify workflows.

* Operator Control Panel - Web-based UI designed for operators to monitor workflows in action, launch workflows manually, or view reports on workflow actions.

BMC BladeLogic Database Automation: Excludes the use of the BMC Database Automation agent for multiple operations. Once the agent is registered within the console and a job has been executed against the target, the Product license has been consumed.

BMC BladeLogic Server Automation - Compliance Module: Excludes use of BMC
BladeLogic agent for server configuration, patching, and provisioning tasks.


BMC BladeLogic Server Automation Suite: Excludes the use of BMC BladeLogic agent for the application packaging and deployment of internally built, proprietary, or custom-developed code.

BMC Capacity Management for Mainframes: Any BMC Capacity Management for Mainframes Product and/or any BMC Performance Analyzer for Mainframes, BMC Performance Predictor for Mainframes, BMC Performance Perceiver for Mainframes, BMC Performance Analyzer for Mainframe Applications and other related products that may be released as part of the BMC Capacity Management for Mainframes must be licensed for all Computer(s) within the mainframe environment for which the Product or one of its components will process data or execute functionality on behalf of, regardless of whether the Product or one of its components is specifically installed on that Computer. The Products may be installed on or moved to any Computer(s) included in the licensed environment.

BMC Capacity Management Products: Any BMC Capacity Management product for distributed systems environments can be reassigned over time from one Computer to another provided that no data related to the first Computer remains stored in the Capacity Management Product repositories.

BMC Cloud Lifecycle Management - Core License Add-on ("CLM Core"): 
* The Product may only be used in a Cloud Environment.
* The Product includes the right to use BMC Network Automation for the network devices in the Cloud Environment as long as the number of supported Network Devices does not exceed the Licensed Capacity. The Product includes the right to use BMC Network Automation only in order to enable the initial provisioning, on-going network operations, and use of the Virtual Data Center feature for Network Devices in the Cloud Environment. The Product does not include the right to use BMC Network Automation for the management of Network Devices that are not in a BMC Network Automation Pod.
* The Product includes the right to use BMC Atrium Orchestrator for the Licensed Capacity, only in order to deliver the process orchestration use cases that are installed out-of-the-box. The Product does not include the right to use any other functional capabilities or product components, including, but not limited to peers, adapters, and runbook content.

BMC Cloud Lifecycle Management - Standard Pack License Add-on ("CLM Standard"): 
* The Product may only be used in a Cloud Environment.
* If Customer is also a Cloud Service Provider then Product cannot be used by the Cloud Service Provider for other environments, including but not limited to the Cloud Service Provider’s internal IT environment, or System Integration activities for Clients which are not part of Cloud Services. The Product may not be installed on Client premises or accessed or used directly by Clients.
* The Product includes expanded license rights for BMC Atrium Orchestrator including unlimited peer licenses, use of all generally
available Base Adapters, and Development Studio and Operator Control Panel user licenses to support the Licensed Capacity. The Product does not include the right to use any other Application Adapters or Runbook content other than what is installed out-of-the-box.

* Customer may only use the BMC Remedy Service Request Management functionality of the BMC Remedy ITSM product. The Product includes the right to use BMC Remedy Service Request Management for any number of users, to support any service requests that are directly related to the delivery or consumption of Cloud Services, for the Licensed Capacity.

* The Product includes the right to use BMC Network Automation for the network devices in the Cloud Environment as long as the number of supported Network Devices does not exceed the Licensed Capacity. The Product includes the right to use BMC Network Automation only in order to enable the initial provisioning, on-going network operations, and use of the Virtual Data Center feature for Network Devices in the Cloud Environment. The Product does not include the right to use BMC Network Automation for the management of Network Devices that are not in a BMC Network Automation Pod.

* The Product includes the right to use BMC Atrium Orchestrator for the Licensed Capacity, only in order to deliver the process orchestration use cases that are installed out-of-the-box. The Product does not include the right to use any other functional capabilities or product components, including, but not limited to peers, adapters, and runbook content.

* The Product includes the right to use BMC ProactiveNet Performance Management for the Licensed Capacity, only in order to deliver the CPU monitoring capabilities that are installed out-of-the-box. The Product does not include the right to use any other functional capabilities of BMC ProactiveNet Performance Management, including but not limited to, use of the BMC ProactiveNet Performance Management console for operational purposes, monitoring of IT infrastructure beyond what is specified, and any other analytics, diagnostics, event or impact management capabilities.

* The product includes the right to use BMC TrueSight Capacity Optimizer for Servers for the Licensed Capacity only in order to enable the out-of-the-box Capacity Aware Placement Advice capability as part of the CLM Resource Manager. The Product does not include the right to use any other functional capabilities of BMC TrueSight Capacity Optimization, including but not limited to, the use of BMC TrueSight Capacity Optimization for capacity planning, virtualization and consolidation; capacity analysis, forecasting, reporting and dashboards; and capacity metering for showback or chargeback.

**BMC Cloud Lifecycle Management – Foundation Standard Pack License Add-on ("CLM Foundation"):**

* The Product may only be used in a Cloud Environment.

* If Customer is also a Cloud Service Provider then The Product cannot be used by the Cloud Service Provider for other environments, including but not limited to the Cloud Service Provider’s internal IT environment, or System Integration activities for Clients which are not part of Cloud Services. The Product may not be installed on Client premises or accessed or used directly by Clients.

* The Product includes expanded license rights for BMC Atrium Orchestrator including unlimited peer licenses, use of all generally available Base Adapters, and Development Studio and Operator Control
Panel user licenses to support the Licensed Capacity. The Product does not include the right to use any other Application Adapters or Runbook content other than what is installed out-of-the-box.

* The Product includes the right to use the BMC BladeLogic Server Automation (BBSA) for Server Provisioning and Software Deployment functionality only. The Product does not include the right to use any other functionality of BBSA, including, but not limited to, patching, compliance, application release automation, configuration management, discovery, inventory, and nsh-scripting.

* Customer may only use the BMC Remedy Service Request Management functionality of the BMC Remedy ITSM product. The Product includes the right to use BMC Remedy Service Request Management for any number of users, to support any service requests that are directly related to the delivery or consumption of Cloud Services, for the Licensed Capacity.

* The Product includes the right to use BMC Network Automation for the network devices in the Cloud Environment as long as the number of supported Network Devices does not exceed the Licensed Capacity. The Product includes the right to use BMC Network Automation only in order to enable the initial provisioning, on-going network operations, and use of the Virtual Data Center feature for Network Devices in the Cloud Environment. The Product does not include the right to use BMC Network Automation for the management of Network Devices that are not in a BMC Network Automation Pod.

* The Product includes the right to use BMC ProactiveNet Performance Management for the Licensed Capacity, only in order to deliver the CPU monitoring capabilities that are installed out-of-the-box. The Product does not include the right to use any other functional capabilities of BMC ProactiveNet Performance Management, including but not limited to, use of the BMC ProactiveNet Performance Management console for operational purposes, monitoring of IT infrastructure beyond what is specified, and any other analytics, diagnostics, event or impact management capabilities.

* The product includes the right to use BMC Capacity Optimization for the Licensed Capacity only in order to enable the out-of-the-box Capacity Aware Placement Advice capability as part of the CLM Resource Manager. The Product does not include the rights to use any other functional capabilities of BMC Capacity Optimization, including but not limited to, the use of BMC Capacity Optimization for capacity planning, virtualization and consolidation; capacity analysis, forecasting, reporting and dashboards; and capacity metering for showback or chargeback.

License Restriction for Borland Silk Performer Synthetic Transaction Monitoring for BMC Software: The Borland Silk Performer Synthetic Transaction Monitoring for BMC Software product may only be used with BMC performance management products and BMC application performance management products.

Desktop/Mobile Management Product Restrictions for BMC Configuration Management Products: Each "Desktop/Mobile Management" License is limited for use with one Client Endpoint. "Client Endpoint" means a laptop, desktop or other non-Server Computer.

* Desktop/Mobile Patch Management Restriction: A "Desktop/Mobile Patch Management" License may only be used to manage, deploy, update and inventory anti-virus software and security patches on one Client Endpoint.
* Desktop/Mobile Patch Management Pack Restriction: The Desktop/Mobile Application Management Product and the Desktop/Mobile Configuration Discovery Product that are shipped with the Desktop/Mobile Patch Management Pack License may only be used to manage, deploy, update and inventory anti-virus software and security patches on one licensed Client Endpoint, unless Customer has separately licensed the Desktop/Mobile Application Management Product and the Desktop/Mobile Configuration Discovery Product. Customer may not use the functionality of such Products for any other purpose.

* BMC Configuration Management Desktop OS Management Restriction: A "BMC CM Desktop OS Management" License may only be used to manage operating system migration activities on one Client Endpoint. Each BMC CM Desktop OS Management License: (a) may only be used on a licensed Client Endpoint that is licensed for use with both a Desktop/Mobile Application Management License and a Desktop/Mobile Configuration Discovery License; and (b) may not be redeployed or harvested to a different Client Endpoint.

* Extranet Application Management Restriction: An "Extranet Application Management" License may only be used on one Client Endpoint. The parties must mutually agree on the name of each Single Application and its primary function at the time of Order. Single Application is defined as a Tuner channel containing one application with one primary function, and Tuner is defined as is the client component of the Product configured by Customer for deployment on licensed Endpoints.

* BMC Configuration Management Control Center Module Restriction for BMC Configuration Management Products: Each "BMC CM Control Center" License may only be used by Administrators for the project for which it was licensed. An Administrator is defined as an employee with access to or the right to use the administrative components of the Product.

* BMC Configuration Management Developers Kit Definition and Restriction for BMC Configuration Management Products: A "BMC CM Developers Kit" license allows Customer to embed the "SDK Run Time Code" in unmodified object code form, into a single software application developed by Customer to create an "SDK Client." "SDK Run Time Code" means the unmodified object code files in the BMC CM Product that are designated as re-distributable. "SDK Client" means a software technology with a principal purpose and functionality substantially different than that of the SDK Run Time Code and that uses only a BMC Desktop/Mobile Management Product, a BMC Device Management Product and/or a BMC Server Management Product, as applicable, to invoke the update functionality of the SDK Run Time Code. An SDK Client may only be used on, or distributed to, licensed Endpoints that are licensed separately by Customer, which licensed Endpoints may be within or outside of Customer's organization. "Client Endpoint" means a laptop, desktop or other non-Server Computer. "Device Endpoint" means a personal digital assistant or similar computing device. "Endpoint" means a Client Endpoint, a Device Endpoint, a Server Endpoint, or Other Endpoint, as the case may be. "Other Endpoint" means a router, a switch, a hub, or other network device, peripheral or hardware instrument, as the case may be. "Server Endpoint" is any virtual or physical Computer that provides a service for other Computers or users connected to it via the Internet, extranet, intranet, or other networked technologies.

* Device Management Product Restriction for BMC Configuration Management Products: Each "Device Management" License is limited for use with one
Device Endpoint. "Device Endpoint" means a personal digital assistant or similar computing device.

Server Management Product Restrictions for BMC Configuration Management Products: Each "Server Management" License is limited for use per CPU - Subcapacity.

* Server Patch Management Restriction: A "Server Patch Management" License may only be used to manage, deploy, update and inventory anti-virus software and security patches per CPU - Subcapacity.

* Server Patch Management Pack Restriction: The Desktop/Mobile Application Management Product and the Desktop/Mobile Configuration Discovery Product that are shipped with the Server Patch Management Pack License may only be used to manage, deploy, update and inventory anti-virus software and security patches on licensed Server Endpoints, unless Customer has separately licensed the Desktop/Mobile Application Management Product and the Desktop/Mobile Configuration Discovery Product. Customer may not use the functionality of such Products for any other purpose. With respect to the above Server Management Licenses, Customer must comply with any restrictions designated at the time of Order on the maximum number of CPUs that may be included in each Server Endpoint. "Server Endpoint" is any virtual or physical Computer that provides a service for other Computers or users connected to it via the Internet, extranet, intranet, or other networked technologies.

CONTROL-M/Assist: Control-M/Assist may only be used to interface with the third party scheduler and may not be used to schedule or manage batch processes outside of the cross-scheduler dependencies.

Control-M Self Service Mobile Integration Kit: The Control-M Self Service Mobile Integration Kit is governed by the terms and conditions of the license agreement provided with the product.

BMC Database Automation - License Add-on: Excludes the use of the BMC Database Automation agent for multiple operations. Once the agent is registered within the console and a job has been executed against the target, the Product license has been consumed.

BMC Decision Support - Database Automation (5 Viewer/Query Licenses): Excludes use of Report Authoring module which must be licensed separately.


BMC Decision Support - Server Automation (5 Viewer, 1 Query License): Excludes use of Report Authoring module which must be licensed separately.

BMC FootPrints Service Core 5 Named User BASE Software Package: May be used with no more than three workspaces per instance and by no more than 15 named users.

BMC Identity Products:

* Internal User: If a Product name includes the term "Internal User," that Product can only be used by Customer's employees (full time and part time) and contractors whose information is being managed using the BMC IdM tools. Information on these users will typically be found in the HR database.

* External User: If a Product name includes the term "External User," that Product can only be used by Customer's business partners and customers/ prospects whose information is being managed using the BMC
IdM tools or Customer's employees (full or part time)/contractors who are licensed to use one or more of the following BMC Identity Management Tools: (1) BMC Identity User Administration (2) BMC Identity Password Management (3) BMC Identity Compliance Manager, provided the users have no more than 2 logons (access points) being managed by the IdM tools.

* Archive User: If a Product name includes the term "Archive User," that Product can only be used by users whose identity information is stored within the IdM system but is not being actively managed; the information could be stored for the purpose of audit/ forensics etc.

* Developer User: If a Product name includes the term "Developer User," that Product can only be used by users who create or modify applications using the BMC Directory Management Studio.

License Allocation Restriction for BMC ITSM Products: Notwithstanding anything to the contrary in this Order or the Agreement, and when the Product is licensed (i) on the "per named user" Unit of Measurement, Customer may exchange 5 named user licenses of the Product for 2 licenses of the same Product with the Unit of Measurement of "per concurrent user" or (ii) on the "per concurrent user" Unit of Measurement, Customer may exchange 2 concurrent user licenses for 5 licenses of the same Product with the Unit of Measurement of "per named user". Customer will not incur additional charges for such exchange of licenses so long as Customer does not exceed the Licensed Capacity of the Product granted to Customer, however, Customer must: (i) notify BMC in writing of its intent to exchange the Product Licenses and (ii) enter into a separate Order to reflect Customer's new Licensed Capacity and the new Unit of Measurement.

BMC Middleware Management - Performance and Availability and BMC Middleware Management - Transaction Monitoring Products: If Customer is using BMC Middleware Management version 6.0 or lower, notwithstanding the Licensed Capacity the number of individual employees or contractors of Customer to whom access to the Management Console is granted ("End Users") is limited to the lesser of (a) the Licensed Capacity of the Product if priced on a per named user Unit of Measurement, or (b) 250 End Users.

BMC Middleware and Transaction Management Products: If Customer is using BMC Middleware Management version 6.0 or lower, when licensed on the per CPU - Subcapacity or per MIPS Unit of Measurement, regardless of the Licensed Capacity, Customer is limited to ten (10) individual employees or contractors to whom access to the Management Console is granted ("End Users"). When licensed on the per named user Unit of Measurement, Customer is limited to the lesser of (a) the Licensed Capacity of the Product, or (b) 250 End Users.

BMC Mobile Device Management Products. For BMC Mobile Device Management (MDM) Products, any clickwrap agreement with AirWatch, LLC contained in the Products is void and of no effect. Customer's use of these Products is governed by the Agreement.

BMC Monitoring Only Products: Customer is not entitled to use analytics as further detailed in the Documentation.

BMC myIT Suite: Each named user shall be licensed for a maximum of three (3) Device Endpoints. "Device Endpoint" means a personal digital assistant or similar computing device. An additional user license is required for every 3 devices registered by a unique named user. For users who register a fourth device and for every multiple of 3 devices registered by a unique user, an additional license is consumed. When a
device is unregistered and the total number of devices registered by a
unique user falls below 4 or a multiple of 3 devices, a MyIT user license
is released for reuse.

BMC Real End User Experience Monitoring and Analytics - Licensed Add-on
Product: For synthetic transaction monitoring solutions, each instance of
the execution server installed should be counted. For real end user
transaction monitoring solutions, each instance of the watchpoint created
should be counted.

BMC Remedy Products: Customer may not bypass, in any way, the use of a
concurrent or named user license to manage an update (including, without
limitation, submitting a ticket to a parallel form and then using
workflow to perform the update without a license). Concurrent users are
for use within one production instance (a Server or Server group) and may
not be used in more than one production instance.

Development License Restriction for BMC Remedy Products: If a Product
name includes the term "Dev Ln", Customer will restrict installation,
access and use of such Product to a server dedicated to development and
testing only, and will not allow any production or commercial activity on
that server.

Hot Backup License Definition and Restriction for BMC Remedy Products: A
hot backup license is a replicate of the Remedy production licenses on
one backup server. Customer may access that backup server only when the
customary server on which the AR System is installed fails or in
preparation of that backup server for such situation.

Load Balanced System Restriction for BMC Remedy Products: If Customer has
multiple servers in a single logical environment pointing to a single AR
System database instance, only one Instance of Remedy "per Instance"
licenses is required for installation on these servers (except for the AR
System, which must be licensed for each server).

BMC Server Automation - Compliance License Add-on: Excludes use of BMC
BladeLogic agent for server configuration, patching, and provisioning
tasks.

BMC Server Automation - Configuration License Add-on: Excludes use of BMC
BladeLogic agent for server compliance, remediation, and provisioning
tasks.

BMC Server Automation - License Add-on: Excludes the use of BMC
BladeLogic agent for the application packaging and deployment of
internally built, proprietary, or custom-developed code.

BMC Server Automation - Patch License Add-on: Excludes use of BMC
BladeLogic agent for server compliance, remediation, and provisioning
tasks.

BMC Server Automation - Provisioning License Add-on: Excludes use of BMC
BladeLogic agent for server compliance, remediation, and provisioning
tasks.

BMC Service Desk Express Products: No terms in any Business Objects or
Crystal license agreement embedded in the Product apply to the Product.
Customer may make and operate 2 additional copies of the Product solely
for internal pre-production configuration and testing purposes.

BMC Service Desk Express Suite Restriction for BMC Service Desk Express
Products: When purchasing Concurrent User licenses for the "Service Desk
Express" Product, regardless of the number of such licenses purchased and
regardless of the number of purchases made (including future purchases),
Customer is restricted via license keys to a total of (i) five Concurrent
Users conducting a process in the report environment of the Crystal
Reports "Web Server" product which is embedded in the "Service Desk Express" Product and (ii) two named users accessing the "Crystal Reports Professional " product which is bundled with the "Service Desk Express" Product.

Products Installed on Customer's Amazon Web Services ("AWS") Cloud Environment:
* In instances where Customer hosts the Product on Customer's AWS cloud environment, Customer is responsible for working with AWS to ensure the security of its overall AWS cloud environment and the Product.
* An Amazon Machine Image ("AMI") is a special type of pre-configured operating system and virtual application software which is used to create a virtual machine within the Amazon Elastic Compute Cloud ("EC2"). An AMI serves as the basic unit of deployment for services delivered using EC2. In instances where BMC provides Customer with an AMI of a Product to be used on Customer's AWS cloud environment, Customer is responsible for the security of the Product's operation and any potential vulnerabilities in a Product AMI's preconfigured OS stack, Application stack, and/or OS configuration.

c. UNITS OF MEASUREMENT. The following units of measurement apply to certain Products.
per adapter: A license is required for each installation of an adapter that interfaces with the Product.
per agent: A license is required for each unit of software with the official name of Remote Sys Call Daemon or RSCD Agent that can be deployed on a physical or virtual operating system.
per application: A license is required all unique collection of application component templates and configuration objects used to form a single logical platform defined by the Customer.
per asset: A license is required for every physical or logical Server Endpoint, Client Endpoint, Device Endpoint, Data Center Rack, Data Center IP Sensor, or Other Endpoint monitored, managed or discovered by the Product. "Client Endpoint" means a laptop, desktop or other non-Server Computer. "Device Endpoint" means a personal digital assistant or similar computing device. "Other Endpoint" means a router, a switch, a hub, or other network device, peripheral or hardware instrument, as the case may be. "Server Endpoint" is any virtual or physical Computer that provides a service for other Computers or users connected to it via the Internet, extranet, intranet, or other networked technologies.
per Cisco™ UCS Server: A license is required for each Cisco Unified Computing System (UCS) Server on which the Product is installed and/or manages regardless of whether the Product or one of its components is installed on that Server.
per Client Endpoint: A license is required for each Client Endpoint. "Client Endpoint" means a laptop, desktop or other non-Server Computer.
per component: A license is required for all objects that represent a physical or logical part of the service model.
per concurrent access license: A license is required for the maximum number of simultaneous sessions accessing the Product. Sessions are counted in packs of 5.
per concurrent session: A license is required for the maximum number of simultaneous sessions accessing the Product.
per concurrent user: A license is required for the maximum number of individual employees or contractors of Customer to whom simultaneous access has been granted to the Product on a computer or multiple
computers.

per CPU - Full Capacity: A license is required for the total number of active, physical CPUs in each Computer upon which the Product is installed or which the Product manages, either remotely or locally. "CPU" means a physical processor or central unit in a designated Computer containing the logic circuitry that performs the instructions of a Computer's programs and refers to the "socket" which can contain one or more processor cores.

per CPU - Subcapacity: A license is required for all active, physical CPUs which the Product manages, either remotely or locally. "CPU" means a physical processor or central unit in a designated Computer containing the logic circuitry that performs the instructions of a Computer's programs and refers to the "socket" which can contain one or more processor cores.

per database: A license is required for the total allocated database space per host ID or physical Computer which the Product is managing. The total allocated database capacity cannot be segregated or aggregated into lower or higher ranges.

per deployed robot: A license is required for all PATROL End-to-End Response Timer robots deployed.

per Device Endpoint: A license is required for each Device Endpoint. "Device Endpoint" means a personal digital assistant or similar computing device.

per engine: A license is required for each mainframe general purpose engine on the server upon which the Product is installed and/or manages regardless of whether the Product or one of its components is installed on that Server.

per enterprise: A license is required per Customer and per Client each, regardless of the number of times Customer installs the Product in its Enterprise or its Client's Enterprise. "Client" means a third party whose data is processed by Customer and is only permitted if Customer is an authorized BMC service provider.

per gigabyte: A license is required for the total allocated database space of all Computers on which the Product has been installed or operated.

per gigabyte range: A license is required for the total allocated database space per host ID or physical Computer which the Product is managing. The Product may not be moved to another Computer unless the current Computer is taken out of service. The total allocated database capacity cannot be segregated or aggregated into lower or higher ranges among different Computers. For example: if Customer licenses 26-50 gigabytes, the Customer is only licensed for a maximum of 50 gigabytes in total across all the databases of the licensed Product on one particular Computer.

per installed server: A license is required for each Server (with a Classification at the appropriate Tier level, if applicable) upon which the Product or any of its components is installed.

per instance: A license is required for all named occurrences of the Product created or installed in the Enterprise.

per Linux engine: A license is required for all engines of a mainframe Computer on which Customer is running Linux, when applicable classified by Linux Group using BMC's standard Computer classification.

per managed asset - server endpoint: A license is required for every Server Endpoint monitored, managed (directly or indirectly), or
discovered by the Product(s). "Server Endpoint" is any virtual or
physical Computer that provides a service for other Computers or users
connected to it via the Internet, extranet, intranet, or other networked
technologies.

per managed asset - device endpoint: A license is required for every
Device Endpoint that is monitored, managed, or discovered by the
Product(s). A "Device Endpoint" can be any virtual or physical Non-Server
Client Computer (e.g. laptop, desktop computer, PDA, smartphone); any
Network device (e.g. router, switch, hub) standalone or chassis-based
device/card/processor using a unique-IP address (also includes virtual
network devices managed through the IP address of its physical host); and
independent Storage (e.g. a disk array, a fiber switch, a tape library, a
switch director). When applicable, the license must be computed at the
appropriate tier level.

per managed component: A license is required for all objects that
represent a physical or logical part of the service model managed by the
Product.

per managed network device: A license is required for each Network Device
managed using a unique IP-address. "Network Device" means a standalone or
chassis-based network device/card/processor.

per managed server: A license is required for each Server managed by the
Product or one of its components whether locally or remotely. When
applicable, this license must be computed at the appropriate tier level
based on the cumulative count of managed servers. Network Devices are not
counted as Servers. This license does not include the Product's
installation on or management of Integrated Facility for Linux (IFL)
engines. "Network Device" means a standalone or chassis-based network
device/card/processor.

per MIPS: A license is required for the total aggregate number of MIPS
for each Computer, including all Computers coupled in a parallel Sysplex
environment, upon which the Product is installed, or which is managed or
monitored by the Product. MIPS Rating is the aggregate computing power
(expressed in millions of instructions per second) of a Computer, using
the MIPS rating set forth in the then current Gartner Group Rating Guide.
Computer-specific passwords will be issued for the Product.

per monitored element: A license is required for all remotely monitored
elements, such as a Server, database, operating system, URL, firewall,
storage, or network device.

per monitored server: A license is required for each Server (with a
Classification at the appropriate Tier level, if applicable) which the
Product or one of its components is monitoring regardless of whether the
Product is monitoring it locally or remotely.

per named user: A license (with a Classification at the appropriate
Level, if applicable) is required for all individual employees or
contractors or clients of Customer to whom access has been granted to the
Product on a computer or multiple computers typically via the issuance of
a unique ID regardless of whether the individual is actively using the
Product at any given time.

per node: A license is required for every Node which the Product manages
and/or monitors. "Node" means a laptop, desktop, or any virtual or
physical Computer that provides a service for other Computers or users
connected to it via the Internet, extranet, intranet, or other networked
technologies.

per port: A license is required for each port. A port is defined as a
physical connection point used by a storage device to connect other devices or systems. For the purpose of BMC licensing, all active ports (Fibre Channel, iSCSI, etc.) for all managed devices (storage arrays, filers, tape libraries, etc.) are counted. Ports on hosts, gateways and switches are not to be counted.

per project: A license is required for each specific project, facility or business unit, as the case may be specified at the time of order.

per Server Endpoint: A license is required for each Server Endpoint. "Server Endpoint" is any virtual or physical Computer that provides a service for other Computers or users connected to it via the Internet, extranet, intranet, or other networked technologies.

per Service Management MIPS: A license is required for the total aggregate number of MIPS for each Computer, including all Computers coupled in a parallel Sysplex environment, upon which the Product is installed, managed or monitored. MIPS Rating is the aggregate computing power (expressed in millions of instructions per second) of a Computer, using the MIPS rating set forth in the then current Gartner Group Rating Guide.

per site: A license is required for the physical site at which the Product is installed regardless of the number of times the Product is installed.

per task: For all Control-M Products, except those that run exclusively in the Mainframe environment, a license is required for the maximum number of Tasks (as defined below) present in the Control-M "Active Jobs" databases in any 24-hour period, regardless of whether the Tasks execute or not. For the Control-M Products that run exclusively in the Mainframe environment, a license is required for the maximum number of Tasks (as defined below) present in solely the Mainframe environment's Control-M "Active Jobs" database. Tasks in the Control-M "Active Jobs" databases include all Tasks in all Distributed Systems and/or Mainframe environments in any 24-hour period (including but not limited to development, staging, QA, pre-production, production, and test environments), except that, (i) SMART folders/table and sub-folders/tables which contain scheduling definitions and are listed as tasks in the "Active Jobs" databases are not counted as Tasks, (ii) Tasks that have time zone settings may remain in the "Active Jobs" databases for up to three consecutive days, but are only counted as one Task, (iii) a Task that runs more than once during the day (with the same Order ID) is counted as one Task - this includes Tasks that are rerun and cyclic Tasks, and (iv) Tasks that are provided for by licenses under alternative Units of Measurement (i.e. tier or MIPS) are not considered Tasks under this "per task" unit of measurement. The number of steps or scripts executed within the named Task shall have no bearing upon the number of Tasks licensed. "Task" is interchangeable with "job" and means an executable command containing the name of the JCL, CL, DCL, ECL, script or dummy processes that is scheduled to execute, as well as the scheduling criteria, flow control, and resource usage.

per terabyte: A license is required for the total aggregate storage capacity in the Enterprise.

per third-party software: A license is required for each installation of the third-party software product that interfaces with the Product.

YOU AGREE THAT YOU HAVE READ THIS AGREEMENT AND INTEND TO BE BOUND, AS IF YOU HAD SIGNED THIS AGREEMENT IN WRITING. IF YOU ARE ACTING ON BEHALF OF
AN ENTITY, YOU WARRANT THAT YOU HAVE THE AUTHORITY TO ACCEPT THE TERMS OF
THIS AGREEMENT FOR SUCH ENTITY.

DOM4J
Copyright 2001-2010 © MetaStuff, Ltd. All Rights Reserved

Copyright 2001-2010 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation
("Software"), with or without modification, are permitted provided
that the following conditions are met:

1. Redistributions of source code must retain copyright
   statements and notices. Redistributions must also contain a
   copy of this document.

2. Redistributions in binary form must reproduce the
   above copyright notice, this list of conditions and the
   following disclaimer in the documentation and/or other
   materials provided with the distribution.

3. The name "DOM4J" must not be used to endorse or promote
   products derived from this Software without prior written
   permission of MetaStuff, Ltd. For written permission,
   please contact dom4j-info@metastuff.com.

4. Products derived from this Software may not be called "DOM4J"
   nor may "DOM4J" appear in their names without prior written
   permission of MetaStuff, Ltd. DOM4J is a registered
   trademark of MetaStuff, Ltd.

5. Due credit should be given to the DOM4J Project -
   http://dom4j.sourceforge.net

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS
`'AS IS'` AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
Dojo

Copyright © 2005-2009, The Dojo Foundation; All rights reserved

Copyright © 2005-2009, The Dojo Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

JDOM

Copyright © 2000-2004 Jason Hunter & Brett McLaughlin; All rights reserved

Copyright © 2000-2004 Jason Hunter & Brett McLaughlin.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions, and the disclaimer that follows
   these conditions in the documentation and/or other materials
provided with the distribution.

3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact (request_AT_jdom_DOT_org).

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management (request_AT_jdom_DOT_org).

In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following:

"This product includes software developed by the JDOM Project http://www.jdom.org/.
Alternatively, the acknowledgment may be graphical using the logos available at http://www.jdom.org/images/logos.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JDOM AUTHORS OR THE PROJECT CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the JDOM Project and was originally created by Jason Hunter (jhunter_AT_jdom_DOT_org) and Brett McLaughlin (brett_AT_jdom_DOT_org). For more information on the JDOM Project, please see see http://www.jdom.org/.

M2Crypto

Copyright © 2008-2011 Heikki Toivonen; All rights reserved

Copyright © 1999-2004 Ng Pheng Siong. All rights reserved.
Portions copyright © 2004-2006 Open Source Applications Foundation.
All rights reserved.
Portions copyright © 2005-2006 Vrije Universiteit Amsterdam.
All rights reserved.
Copyright © 2008-2010 Heikki Toivonen. All rights reserved.
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in
supporting documentation.

THE AUTHOR PROVIDES THIS SOFTWARE "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

PLY

Copyright © 2001-2011 David M. Beazley

PLY (Python Lex-Yacc) Version 3.4

Copyright © 2001-2011,
David M. Beazley (Dabeaz LLC)
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the David Beazley or Dabeaz LLC may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Python

Copyright © 2001-2014 Python Software Foundation; All rights reserved
We apply patches to the Python 2.7.1 source to accomplish the following:

- Stop building of the bsddb module as we use the additional module separately
- Fix for http://bugs.python.org/issue1692335
- Replace calls from select to poll in the select module to
  overcome limitations on maximum file descriptor number

This is the official license for the Python 2.7 release:

A. HISTORY OF THE SOFTWARE

Python was created in the early 1990s by Guido van Rossum at Stichting
Mathematisch Centrum (CWI, see http://www.cwi.nl) in the Netherlands
as a successor of a language called ABC. Guido remains Python's
principal author, although it includes many contributions from others.

In 1995, Guido continued his work on Python at the Corporation for
National Research Initiatives (CNRI, see http://www.cnri.reston.va.us)
in Reston, Virginia where he released several versions of the
software.

In May 2000, Guido and the Python core development team moved to
BeOpen.com to form the BeOpen PythonLabs team. In October of the same
year, the PythonLabs team moved to Digital Creations (now Zope
Corporation, see http://www.zope.com). In 2001, the Python Software
Foundation (PSF, see http://www.python.org/psf/) was formed, a
non-profit organization created specifically to own Python-related
Intellectual Property. Zope Corporation is a sponsoring member of
the PSF.

All Python releases are Open Source (see http://www.opensource.org for
the Open Source Definition). Historically, most, but not all, Python
releases have also been GPL-compatible; the table below summarizes
the various releases.

<table>
<thead>
<tr>
<th>Release</th>
<th>Derived from</th>
<th>Year</th>
<th>Owner</th>
<th>GPL-compatible? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9.0 thru 1.2</td>
<td></td>
<td>1991-1995</td>
<td>CWI</td>
<td>yes</td>
</tr>
<tr>
<td>1.3 thru 1.5.2</td>
<td>1.2</td>
<td>1995-1999</td>
<td>CNRI</td>
<td>yes</td>
</tr>
<tr>
<td>1.6</td>
<td>1.5.2</td>
<td>2000</td>
<td>CNRI</td>
<td>no</td>
</tr>
<tr>
<td>2.0</td>
<td>1.6</td>
<td>2000</td>
<td>BeOpen.com</td>
<td>no</td>
</tr>
<tr>
<td>1.6.1</td>
<td>1.6</td>
<td>2001</td>
<td>CNRI</td>
<td>yes (2)</td>
</tr>
<tr>
<td>2.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>no</td>
</tr>
<tr>
<td>2.0.1</td>
<td>2.0+1.6.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.1</td>
<td>2.1+2.0.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2</td>
<td>2.1.1</td>
<td>2001</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.2</td>
<td>2.1.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.1.3</td>
<td>2.1.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>Version</td>
<td>Parent</td>
<td>Year</td>
<td>License</td>
<td>Status</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>2.2.1</td>
<td>2.2</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.2</td>
<td>2.2.1</td>
<td>2002</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.2.3</td>
<td>2.2.2</td>
<td>2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3</td>
<td>2.2.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.1</td>
<td>2.3</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.2</td>
<td>2.3.1</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.3</td>
<td>2.3.2</td>
<td>2002-2003</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.4</td>
<td>2.3.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.3.5</td>
<td>2.3.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4</td>
<td>2.3</td>
<td>2004</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.1</td>
<td>2.4</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.2</td>
<td>2.4.1</td>
<td>2005</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.3</td>
<td>2.4.2</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.4.4</td>
<td>2.4.3</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5</td>
<td>2.4</td>
<td>2006</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5.1</td>
<td>2.5</td>
<td>2007</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5.2</td>
<td>2.5.1</td>
<td>2008</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.5.3</td>
<td>2.5.2</td>
<td>2008</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6</td>
<td>2.5</td>
<td>2008</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6.1</td>
<td>2.6</td>
<td>2008</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6.2</td>
<td>2.6.1</td>
<td>2009</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6.3</td>
<td>2.6.2</td>
<td>2009</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6.4</td>
<td>2.6.3</td>
<td>2009</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.6.5</td>
<td>2.6.4</td>
<td>2010</td>
<td>PSF</td>
<td>yes</td>
</tr>
<tr>
<td>2.7</td>
<td>2.6</td>
<td>2010</td>
<td>PSF</td>
<td>yes</td>
</tr>
</tbody>
</table>

Footnotes:

(1) GPL-compatible doesn't mean that we're distributing Python under the GPL. All Python licenses, unlike the GPL, let you distribute a modified version without making your changes open source. The GPL-compatible licenses make it possible to combine Python with other software that is released under the GPL; the others don't.

(2) According to Richard Stallman, 1.6.1 is not GPL-compatible, because its license has a choice of law clause. According to CNRI, however, Stallman's lawyer has told CNRI's lawyer that 1.6.1 is "not incompatible" with the GPL.

Thanks to the many outside volunteers who have worked under Guido's direction to make these releases possible.

B. TERMS AND CONDITIONS FOR ACCESSING OR OTHERWISE USING PYTHON

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.
2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on or incorporates Python or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS" basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee. This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee agrees to be bound by the terms and conditions of this License Agreement.

BEOPEN.COM LICENSE AGREEMENT FOR PYTHON 2.0
---------------------------------------------

BEOPEN PYTHON OPEN SOURCE LICENSE AGREEMENT VERSION 1

1. This LICENSE AGREEMENT is between BeOpen.com ("BeOpen"), having an office at 160 Saratoga Avenue, Santa Clara, CA 95051, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. Subject to the terms and conditions of this BeOpen Python License
Agreement, BeOpen hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the BeOpen Python License is retained in the Software, alone or in any derivative version prepared by Licensee.

3. BeOpen is making the Software available to Licensee on an "AS IS" basis. BEOPEN MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, BEOPEN MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. BEOPEN SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

5. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. This License Agreement shall be governed by and interpreted in all respects by the law of the State of California, excluding conflict of law provisions. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between BeOpen and Licensee. This License Agreement does not grant permission to use BeOpen trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party. As an exception, the "BeOpen Python" logos available at http://www.pythonlabs.com/logos.html may be used according to the permissions granted on that web page.

7. By copying, installing or otherwise using the software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

CNRI LICENSE AGREEMENT FOR PYTHON 1.6.1
---------------------------------------

1. This LICENSE AGREEMENT is between the Corporation for National Research Initiatives, having an office at 1895 Preston White Drive, Reston, VA 20191 ("CNRI"), and the Individual or Organization ("Licensee") accessing and otherwise using Python 1.6.1 software in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, CNRI hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python 1.6.1 alone or in any derivative version, provided, however, that CNRI's License Agreement and CNRI's notice of copyright, i.e., "Copyright (c) 1995-2001 Corporation for National Research Initiatives; All Rights
Reserved" are retained in Python 1.6.1 alone or in any derivative version prepared by Licensee. Alternately, in lieu of CNRI's License Agreement, Licensee may substitute the following text (omitting the quotes): "Python 1.6.1 is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python 1.6.1 may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1013. This Agreement may also be obtained from a proxy server on the Internet using the following URL: http://hdl.handle.net/1895.22/1013".

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 1.6.1 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 1.6.1.

4. CNRI is making Python 1.6.1 available to Licensee on an "AS IS" basis. CNRI MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, CNRI MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 1.6.1 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. CNRI SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 1.6.1 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 1.6.1, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. This License Agreement shall be governed by the federal intellectual property law of the United States, including without limitation the federal copyright law, and, to the extent such U.S. federal law does not apply, by the law of the Commonwealth of Virginia, excluding Virginia's conflict of law provisions. Notwithstanding the foregoing, with regard to derivative works based on Python 1.6.1 that incorporate non-separable material that was previously distributed under the GNU General Public License (GPL), the law of the Commonwealth of Virginia shall govern this License Agreement only as to issues arising under or with respect to Paragraphs 4, 5, and 7 of this License Agreement. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between CNRI and Licensee. This License Agreement does not grant permission to use CNRI trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By clicking on the "ACCEPT" button where indicated, or by copying, installing or otherwise using Python 1.6.1, Licensee agrees to be bound by the terms and conditions of this License Agreement.

ACCEPT
CWI LICENSE AGREEMENT FOR PYTHON 0.9.0 THROUGH 1.2

Copyright (c) 1991 - 1995, Stichting Mathematisch Centrum Amsterdam, The Netherlands. All rights reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Stichting Mathematisch Centrum or CWI not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

STICHTING MATHEMATISCH CENTRUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL STICHTING MATHEMATISCH CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Python win32 extensions

Copyright © 1994-2008, Mark Hammond; All rights reserved

Unless stated in the specific source file, this work is
Copyright © 1994-2008, Mark Hammond
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither name of Mark Hammond nor the name of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS `AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Python-ldap

Copyright © 2008-2014, python-ldap project team

The python-ldap package is distributed under Python-style license.
Standard disclaimer:
This software is made available by the author(s) to the public for free
and "as is". All users of this free software are solely and entirely
responsible for their own choice and use of this software for their
own purposes. By using this software, each user agrees that the
author(s) shall not be liable for damages of any kind in relation to
its use or performance. The author(s) do not warrant that this software
is fit for any purpose.

Red Hat

Copyright © 2003 Red Hat, Inc. All rights reserved

Copyright © 2003 Red Hat, Inc. All rights reserved. "Red Hat" and the Red
Hat "Shadowman" logo are registered trademarks of Red Hat, Inc. "Linux" is
a registered trademark of Linus Torvalds. All other trademarks are the
property of their respective owners.

LICENSE AGREEMENT AND LIMITED PRODUCT WARRANTY
RED HAT® ENTERPRISE LINUX® AND RED HAT® APPLICATIONS

This agreement governs the use of the Software and any updates to the
Software, regardless of the delivery mechanism. The Software is a
collective work under U.S. Copyright Law. Subject to the following terms,
Red Hat, Inc. ("Red Hat") grants to the user ("Customer") a license to
this collective work pursuant to the GNU General Public License.

1. The Software. Red Hat Enterprise Linux and Red Hat Applications (the
"Software") are either a modular operating system or application
consisting of hundreds of software components. The end user license
agreement for each component is located in the component's source code.
With the exception of certain image files identified in Section 2 below,
the license terms for the components permit Customer to copy, modify, and
redistribute the component, in both source code and binary code forms.
This agreement does not limit Customer's rights under, or grant Customer rights that supersede, the license terms of any particular component.

2. Intellectual Property Rights. The Software and each of its components, including the source code, documentation, appearance, structure and organization are owned by Red Hat and others and are protected under copyright and other laws. Title to the Software and any component, or to any copy, modification, or merged portion shall remain with the aforementioned, subject to the applicable license. The "Red Hat" trademark and the "Shadowman" logo are registered trademarks of Red Hat in the U.S. and other countries. This agreement does not permit Customer to distribute the Software using Red Hat's trademarks. Customer should read the information found at http://www.redhat.com/about/corporate/trademark/ before distributing a copy of the Software, regardless of whether it has been modified. If Customer makes a commercial redistribution of the Software, unless a separate agreement with Red Hat is executed or other permission granted, then Customer must modify any files identified as "REDHAT-LOGOS" and "anaconda-images" to remove all images containing the "Red Hat" trademark or the "Shadowman" logo. Merely deleting these files may corrupt the Software.

3. Limited Warranty. Except as specifically stated in this agreement or a license for a particular component, to the maximum extent permitted under applicable law, the Software and the components are provided and licensed "as is" without warranty of any kind, expressed or implied, including the implied warranties of merchantability, non-infringement or fitness for a particular purpose. Red Hat warrants that the media on which the Software is furnished will be free from defects in materials and manufacture under normal use for a period of 30 days from the date of delivery to Customer. Red Hat does not warrant that the functions contained in the Software will meet Customer's requirements or that the operation of the Software will be entirely error free or appear precisely as described in the accompanying documentation. This warranty extends only to the party that purchases the Software from Red Hat or a Red Hat authorized distributor.

4. Limitation of Remedies and Liability. To the maximum extent permitted by applicable law, the remedies described below are accepted by Customer as its only remedies. Red Hat's entire liability, and Customer's exclusive remedies, shall be: If the Software media is defective, Customer may return it within 30 days of delivery along with a copy of Customer's payment receipt and Red Hat, at its option, will replace it or refund the money paid by Customer for the Software. To the maximum extent permitted by applicable law, Red Hat or any Red Hat authorized dealer will not be liable to Customer for any incidental or consequential damages, including lost profits or lost savings arising out of the use or inability to use the Software, even if Red Hat or such dealer has been advised of the possibility of such damages. In no event shall Red Hat's liability under this agreement exceed the amount that Customer paid to Red Hat under this agreement during the twelve months preceding the action.

5. Export Control. As required by U.S. law, Customer represents and warrants that it: (a) understands that the Software is subject to export
controls under the U.S. Commerce Department's Export Administration Regulations ("EAR"); (b) is not located in a prohibited destination country under the EAR or U.S. sanctions regulations (currently Cuba, Iran, Iraq, Libya, North Korea, Sudan and Syria); (c) will not export, re-export, or transfer the Software to any prohibited destination, entity, or individual without the necessary export license(s) or authorizations(s) from the U.S. Government; (d) will not use or transfer the Software for use in any sensitive nuclear, chemical or biological weapons, or missile technology end-uses unless authorized by the U.S. Government by regulation or specific license; (e) understands and agrees that if it is in the United States and exports or transfers the Software to eligible end users, it will, as required by EAR Section 740.17(e), submit semi-annual reports to the Commerce Department's Bureau of Industry & Security (BIS), which include the name and address (including country) of each transferee; and (f) understands that countries other than the United States may restrict the import, use, or export of encryption products and that it shall be solely responsible for compliance with any such import, use or export restrictions.

6. Third Party Programs. Red Hat may distribute third party software programs with the Software that are not part of the Software. These third party programs are subject to their own license terms. The license terms either accompany the programs or can be viewed at http://www.redhat.com/licenses/. If Customer does not agree to abide by the applicable license terms for such programs, then Customer may not install them. If Customer wishes to install the programs on more than one system or transfer the programs to another party, then Customer must contact the licensor of the programs.

7. General. If any provision of this agreement is held to be unenforceable, that shall not affect the enforceability of the remaining provisions. This agreement shall be governed by the laws of the State of North Carolina and of the United States, without regard to any conflict of laws provisions, except that the United Nations Convention on the International Sale of Goods shall not apply.
* Neither the name of the company nor the names of its contributors may
  be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE OFFICERS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

SNMP++
Copyright © 2001-2010 Jochen Katz, Frank Fock

This software is based on SNMP++ 2.6 from Hewlett Packard:

Copyright © 1996
Hewlett-Packard Company

ATTENTION: USE OF THIS SOFTWARE IS SUBJECT TO THE FOLLOWING TERMS.
Permission to use, copy, modify, distribute and/or sell this software
and/or its documentation is hereby granted without fee. User agrees
display the above copyright notice and this license notice in all
copies of the software and any documentation of the software. User
agrees to assume all liability for the use of the software;
Hewlett-Packard and Jochen Katz make no representations about the
suitability of this software for any purpose. It is provided
"AS-IS" without warranty of any kind, either express or implied. User
hereby grants a royalty-free license to any and all derivatives based
upon this software code base.

Tom Sawyer Visualization
Copyright © 2005 Tom Sawyer Software, Oakland, California, All rights reserved

VI Java
Copyright © 2008 VMware, Inc. All Rights Reserved

Copyright (c) 2008 VMware, Inc. All Rights Reserved.
Copyright (c) 2009 Altor Networks. All Rights Reserved.
Copyright (c) 2009 NetApp. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of VMware, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL VMware, Inc. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Webware

Copyright © 1999-2010 by Chuck Esterbrook; All rights reserved

The Gist

Webware for Python is open source, but there is no requirement that products developed with or derivative to Webware become open source.

Webware for Python is copyrighted, but you can freely use and copy it as long as you don't change or remove this copyright notice. The license is a clone of the original OSI-approved Python license.

There is no warranty of any kind. Use at your own risk.

Read this entire document for complete, legal details.

Copyright

Copyright © 1999-2010 by Chuck Esterbrook; All rights reserved.
License

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of the authors not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

Disclaimer

THE AUTHORS DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Trademarks

All trademarks referred to in the source code, documentation, sample files or otherwise, are reserved by their respective owners.

bsddb3

Copyright © 1999-2011, Digital Creations, Fredericksburg, VA, USA and Andrew Kuchling; All rights reserved

Copyright © 1999-2011, Digital Creations, Fredericksburg, VA, USA and Andrew Kuchling; All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions, and the disclaimer that follows.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Digital Creations nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY DIGITAL CREATIONS AND CONTRIBUTORS *AS IS* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL DIGITAL
CREATIONS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS
OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE
USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

ipaddr
Copyright © 2007 Google Inc

ipaddr Copyright © 2007 Google Inc
Licensed to PSF under a Contributor Agreement.
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied. See the License for the specific language governing
permissions and limitations under the License.

http://code.google.com/p/ipaddr-py/

json.py
Copyright © 2005 Patrick D. Logan.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc.
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.
0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a
facility refers to a function or data to be supplied by an Application
that uses the facility (other than as an argument passed when the
facility is invoked), then you may convey a copy of the modified
version:

a) under this License, provided that you make a good faith effort to
ensure that, in the event an Application does not supply the
function or data, the facility still operates, and performs
whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of
this License applicable to that copy.


The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.


You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the
Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)


You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

mxDateTime

EGENIX.COM PUBLIC LICENSE AGREEMENT VERSION 1.1.0
1. Introduction

This "License Agreement" is between eGenix.com Software, Skills and Services GmbH ("eGenix.com"), having an office at Pastor-Loeh-Str. 48, D-40764 Langenfeld, Germany, and the Individual or Organization ("Licensee") accessing and otherwise using this software in source or binary form and its associated documentation ("the Software").

2. License

Subject to the terms and conditions of this eGenix.com Public License Agreement, eGenix.com hereby grants Licensee a non-exclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use the Software alone or in any derivative version, provided, however, that the eGenix.com Public License Agreement is retained in the Software, or in any derivative version of the Software prepared by Licensee.

3. NO WARRANTY

eGenix.com is making the Software available to Licensee on an "AS IS" basis. SUBJECT TO ANY STATUTORY WARRANTIES WHICH CAN NOT BE EXCLUDED, EGENIX.COM MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, EGENIX.COM MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

4. LIMITATION OF LIABILITY

EGENIX.COM SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF THE SOFTWARE FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THE SOFTWARE, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE EXCLUSION OR LIMITATION MAY NOT APPLY TO LICENSEE.

5. Termination

This License Agreement will automatically terminate upon a material breach of its terms and conditions.

6. Third Party Rights

Any software or documentation in source or binary form provided along with the Software that is associated with a separate license
agreement is licensed to Licensee under the terms of that license agreement. This License Agreement does not apply to those portions of the Software. Copies of the third party licenses are included in the Software Distribution.

7. General

Nothing in this License Agreement affects any statutory rights of consumers that cannot be waived or limited by contract.

Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between eGenix.com and Licensee.

If any provision of this License Agreement shall be unlawful, void, or for any reason unenforceable, such provision shall be modified to the extent necessary to render it enforceable without losing its intent, or, if no such modification is possible, be severed from this License Agreement and shall not affect the validity and enforceability of the remaining provisions of this License Agreement.

This License Agreement shall be governed by and interpreted in all respects by the law of Germany, excluding conflict of law provisions. It shall not be governed by the United Nations Convention on Contracts for International Sale of Goods.

This License Agreement does not grant permission to use eGenix.com trademarks or trade names in a trademark sense to endorse or promote products or services of Licensee, or any third party.

The controlling language of this License Agreement is English. If Licensee has received a translation into another language, it has been provided for Licensee’s convenience only.

8. Agreement

By downloading, copying, installing or otherwise using the Software, Licensee agrees to be bound by the terms and conditions of this License Agreement.

For question regarding this License Agreement, please write to:

eGenix.com Software, Skills and Services GmbH
Pastor-Loeh-Str. 48
D-40764 Langenfeld
Germany

omniORB

Copyright © AT&T Laboratories Cambridge, Apasphere Ltd and others
This is omniORB 4.1.6.

omniORB is copyright Apasphere Ltd, AT&T Laboratories Cambridge and others. It is free software. The programs in omniORB are distributed under the GNU General Public Licence as published by the Free Software Foundation. See the file COPYING for copying permission of these programs. The libraries in omniORB are distributed under the GNU Lesser General Public Licence. See the file COPYING.LIB for copying permission of these libraries.

We impose no restriction on the use of the IDL compiler output. The stub code produced by the IDL compiler is not considered a derived work of it.

pexpect

Copyright © 2008 Noah Spurrier

Pexpect Copyright © 2008 Noah Spurrier

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

http://pexpect.sourceforge.net/

portal.js

Copyright © 2006 Sébastien Gruhier


Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright © 2005-2010 Sam Stephenson

Copyright © 2005-2010 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

http://madrobby.github.com/scriptaculous/license/

Other licensing terms
Some portions of this software are covered by the Apache License
Some portions of this software are covered by the Common Development and Distribution License (CDDL)
Some portions of this software are covered by the GNU General Public License (GPL) and/or the GNU Lesser General Public License (LGPL).

All Open Source packages are available from BMC Software directly - visit our support page to request them.

Known and corrected issues
The following list provides links to the defects resolved in the supported versions of BMC Atrium Discovery and the known defects in the current version of the product:

- Defects resolved in this version
- Known defects in this version

Defects resolved in this version

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect No</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRUD1-16505</td>
<td>Problem: The MAC addresses of some HP iLO management controllers are not discovered. Solution: Code fix.</td>
<td>QM001884207</td>
</tr>
<tr>
<td>DRUD1-16502</td>
<td>Problem: Node creation incorrectly reports FoundMultipleMatches. Solution: Code fix.</td>
<td>ISS04532052 ISS04490968</td>
</tr>
</tbody>
</table>
Defects resolved in BMC Atrium Discovery Version 10.2.0.2

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect No</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-16484</td>
<td>Problem: If the CMDB is extremely slow, the CMDB API call to update a batch of CIs can time out.</td>
<td>ISS04513482</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix. See the Batch Size option in Setting up a CMDB synchronization connection.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16464</td>
<td>Problem: If the CMDB sync queue is not empty, you cannot upgrade, or apply an OS update.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16496</td>
<td>Problem: If the system is restarted when an in-progress scheduled scan is being deleted, recovery can fail.</td>
<td>QM001887171</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16494</td>
<td>Problem: Product catalog errors cause CMDB resync to fail, and abort sync of a root device.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16487</td>
<td>Problem: The CMDB sync shadow copy contains live and deleted versions of some nodes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16483</td>
<td>Problem: The filter for BMC_SoftwareServer removes relationships between BMC_SoftwareServer CIs that are not affected by the filter.</td>
<td>QM001884052</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16466</td>
<td>Problem: The CMDB sync filter can construct an expression that cannot be evaluated.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16462</td>
<td>Problem: CMDB sync transformer does not continue after errors in an extension syncmapping.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16460</td>
<td>Problem: Pool data cannot be copied from one system to another and marked as non-expiring. Doing so causes discovery to fail.</td>
<td>QM001885279</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td>ADDM-16454</td>
<td>Problem: CMDB sync recommends a resync more often than is actually required.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16447</td>
<td>Problem: During resync, nodes are created in the shadow partition for any that did not previously exist. If the corresponding creates then fail in the CMDB, for example due to an integrity constraint set on an attribute, shadow partition nodes will exist without corresponding CIs. ADDM may subsequently attempt to update these non-existent CIs or create relationships to them, resulting in messages like &quot;ERROR (120006): Instance not found&quot;, and &quot;ERROR (100): Required Entry ID is empty&quot;. A resync may prevent further occurrences of these messages.</td>
<td>QM001885611</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix. Note that a resync performed after upgrading to 10.2.0.2 should permanently prevent recurrences of the above messages.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15536</td>
<td>Problem: CMDB component filter handles &quot;None&quot; condition incorrectly.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16468</td>
<td>Problem: CMDB Sync filter is broken for missing attributes.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16475</td>
<td>Problem: CMDB component filter handles &quot;None&quot; condition incorrectly.</td>
<td>BMC Communities link</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
</tbody>
</table>
## Defects resolved in BMC Atrium Discovery Version 10.2.0.1

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect No</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-16409</td>
<td>Problem: Communication timeout to the CMDB during resync allows the nodeset to be scavenged, leading to an exception.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16383</td>
<td>Problem: Additions to OSType enum in BMC_OperatingSystem sync mapping cause sync errors.</td>
<td>QM001882821</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16374</td>
<td>Problem: BAD_PARAM exception thrown when XMLAPI getHeuristicData fails.</td>
<td>QM001883473</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16366</td>
<td>Problem: DDD aging failed to start: &quot;Unable to age DiscoveryAccess: No ECA controller&quot;.</td>
<td>QM001882464</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16363</td>
<td>Problem: CMDB resync deletes data created by Normalization Engine in ADDM dataset.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16359</td>
<td>Problem: A resync started immediately after running <code>tw_model_wipe</code> is extremely slow.</td>
<td>QM001882958</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16358</td>
<td>Problem: Cluster rebalance fails on a datastore node larger than 2GB.</td>
<td>QM001860421</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16346</td>
<td>Problem: Resync commit - CI creation fails when a key contains invalid UTF-8 characters.</td>
<td>QM001882049</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16340</td>
<td>Problem: &quot;License Detail&quot; of vCenter Server SI does not contain unassigned licenses.</td>
<td>QM001881844</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16299</td>
<td>Problem: &quot;Internal processing error&quot; during &quot;prepare resync&quot; phase.</td>
<td>QM001881125</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16291</td>
<td>Problem: Software Context button should be displayed, but disabled in unsupported browsers.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16279</td>
<td>Problem: The omniORB Diffie-Hellman parameter size needs to be increased. See LogJam blog post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16272</td>
<td>Problem: Disc space monitor does not shut down services.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16230</td>
<td>Problem: Occasionally a device may fail (null pointer error) to sync to CMDB.</td>
<td>QM001882424</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16186</td>
<td>Problem: Cannot force remove member if that machine is alive.</td>
<td>QM001879993</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15985</td>
<td>Problem: Solaris package errors reported as a single package.</td>
<td>QM001818360</td>
</tr>
<tr>
<td></td>
<td>Solution: Code fix.</td>
<td></td>
</tr>
</tbody>
</table>
# Defects Resolved in BMC Atrium Discovery Version 10.2

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-16159</td>
<td>Problem: Access failures with valid credentials, due to the use of a TERM of Linux which causes failures on some systems.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16137</td>
<td>Problem: SI to BAI relationship removals not syncing to CMDB.</td>
<td>QM001878374</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16136</td>
<td>Problem: Unexpected CMDB sync soft-deletion of BMC_ApplicationSystem when BAI contains SIs on different Hosts.</td>
<td>QM001878228</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16125</td>
<td>Problem: Informix Database report under the infrastructure tab shows &quot;no result&quot; although Informix databases have been discovered.</td>
<td>QM001877956</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16115</td>
<td>Problem: omniORB still supports SSLv3 on appliance.</td>
<td>QM001877831</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16113</td>
<td>Problem: Improve discovery options to avoid null payload to be used for NMAP ICMP packets.</td>
<td>QM001877895</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16074</td>
<td>Problem: Load Balancer Fields module can cause UI error if pools were not found for member.</td>
<td>QM001876243</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16071</td>
<td>Problem: Reasoning Rule to link REMOTE and EXPORTED FileSystem nodes does not handle all cases.</td>
<td>QM001878136</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16060</td>
<td>Problem: UI Performance issues BMC Atrium Discovery 10.1 when using Internet Explorer 9 and 11.</td>
<td>QM001873542</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16016</td>
<td>Problem: UI Crashes on baseline status page for a public group user which has only Baseline read permission.</td>
<td>QM001872995</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15997</td>
<td>Problem: LDAP authentication doesn't support PasswordExpiration LDAP Control.</td>
<td>QM001871666</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15996</td>
<td>Problem: getLDAPErrorString() from ldapauth can't process LDAPException if its argument is a string.</td>
<td>QM001871665</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15982</td>
<td>Problem: SSH Server CBC Mode Ciphers were supported.</td>
<td>QM001872400</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15981</td>
<td>Problem: SSL RC4 Cipher Suites were supported.</td>
<td>QM001872404</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15974</td>
<td>Problem: Killing the watchdog process leaves zombie BMC Atrium Discovery service process and hung watchdog.</td>
<td>QM001870298</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15971</td>
<td>Problem: SSH Weak MAC Algorithms enabled.</td>
<td>QM001872405</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix.</td>
<td>QM001867446</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>
| ADDM-15958 | **Problem:** BMC Atrium Discovery should stop filtering the ITSM companies where type="service provider".  
**Resolution:** Code fix.                                                                                                                                                                       |                 |
| ADDM-15956 | **Problem:** Backup verification can fail if report names contain multiple spaces.  
**Resolution:** Code fix.                                                                                                                                                                       | OM001868972 \ ISS04379110 |
| ADDM-15948 | **Problem:** Permissions on /usr/bin/htpasswd should be 550 at least.  
**Resolution:** Code fix.                                                                                                                                                                       | OM001868988     |
| ADDM-15937 | **Problem:** Zonename is always converted to lowercase though Solaris zone names are case sensitive.  
**Resolution:** Code fix.                                                                                                                                                                       | OM001865816     |
| ADDM-15930 | **Problem:** Creating or editing SNMP recognition rule fails on a device with no IP address.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001871585     |
| ADDM-15911 | **Problem:** Additional path to prtdiag tool needs to be added to Solaris platform script.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001865275     |
| ADDM-15811 | **Problem:** Some pre-scanning errors are logged.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001875509     |
| ADDM-15799 | **Problem:** tw_root_node_key_import should not abort on Host nodes with no MAC address.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001858934     |
| ADDM-15759 | **Problem:** Different best IP addresses reported on scanner/consolidator. A consolidated host is not updated anymore, the IP address is skipped on the consolidator but shows good access on the scanner.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001861990     |
| ADDM-15573 | **Problem:** vCenter credentials cannot be edited if port settings changed.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001849685 \ ISS04316293 |
| ADDM-15566 | **Problem:** ECA error: RuleError on rule link_remote_to_export_on_confirmed.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001848628     |
| ADDM-15419 | **Problem:** Poorly formatted baseline email.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001857806     |
| ADDM-14952 | **Problem:** Export failing for RDBMS mapping with the following error "Unable to extract POJO from Any: the type sequence <sequence <long>> is unsupported".  
**Resolution:** Code fix.                                                                                                                                                                       | QM001805581     |
| ADDM-14190 | **Problem:** CMDB Sync fails with "BAD_PARAM_WrongPythonType" when trying to run a custom syncmapping.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001765769     |
| ADDM-14088 | **Problem:** PDF Group Report shows a cryptic error when non-ascii characters are present in notes field.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001760173     |
| ADDM-13191 | **Problem:** In the case of CMDB sync VM crash, the error message is not user friendly enough.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001703498     |
| ADDM-12440 | **Problem:** Pattern configuration integer cannot be used in pattern search query.  
**Resolution:** Code fix.                                                                                                                                                                       | QM001857787     |
## Defects Resolved in BMC Atrium Discovery Version 10.1 Patch 1

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-15917 | **Problem:** CAM definition with lots of subword tests can generate a regular expression that does not compile.  
**Resolution:** Code fix.                                      | QM001866334   |
| ADDM-15701 | **Enhancement:** Need a way to regenerate all CAM patterns from their definitions. |               |
| ADDM-14432 | **Problem:** Functional Component created in CAM is not updating / releasing database SI.  
**Resolution:** Code fix. The Functional Components now update and release database SIs once the patterns have been regenerated. | QM001778389   |
| ADDM-15877 | **Problem:** Software context: arrow at the wrong end for a BAI containing a SoftwareComponent.  
**Resolution:** Code fix.                                        |               |
| ADDM-15391 | **Problem:** CMDB Component Filter fails to update as node kind is changed.  
**Resolution:** Code fix.                                         | QM001865454   |
| ADDM-15915 | **Problem:** Consolidation can fail with KeyError producing very large tw_svc_reasoning.out.  
**Resolution:** Code fix.                                         | QM001865533   |
| ADDM-15912 | **Problem:** nmap in pre-scanning may fail to identify a host as up.  
**Resolution:** Code fix.                                         | QM001865551   |
| ADDM-15808 | **Problem:** WBEM credential test took 4 hours.  
**Resolution:** Code fix.                                         |               |
| ADDM-15756 | **Problem:** Discovery can fail to discover some Linux machines with error: Failed to run VER.  
**Resolution:** Code fix.                                         | QM001862142   |
| ADDM-15904 | **Problem:** ECA error in load balancer linking due to a search function.  
**Resolution:** Code fix.                                         |               |
| ADDM-15865 | **Problem:** Fields defined in the taxonomy as list:int are reported as invalid if the real value is a list of longs.  
**Resolution:** Code fix.                                         |               |
| ADDM-15837 | **Problem:** Revert to standalone fails if SSL key has a password.  
**Resolution:** Code fix.                                         |               |
| ADDM-15699 | **Problem:** Upgrade from some versions breaks thumbnails and links for videos.  
**Resolution:** Code fix.                                         | QM001860224   |
| ADDM-15886 | **Problem:** “ERROR: Could not record post upgrade message” on consolidator.  
**Resolution:** Code fix.                                         |               |
| ADDM-15920 | **Problem:** Reasoning can fail to upgrade to 10.1.  
**Resolution:** Code fix.                                         | QM001865746   |
| ADDM-15739 | **Problem:** Reasoning can fail to upgrade from 10.0.0.2.  
**Resolution:** Code fix.                                         | QM001864791   |
| ADDM-15185 | **Problem:** Hosts with no DiscoveryAccess and no EndpointIdentity relationships has been observed. This prevents these Hosts to age.  
**Resolution:** Code fix.                                         | QM001816543   |
## Defects Resolved in BMC Atrium Discovery Version 10.1

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-14140 | **Problem:** CMDB sync exporter needs to be updated to handle Itanium processor family.  
**Solution:** Code fix.                     | QM001763938         |
| ADDM-14815 | **Problem:** CAM search engine does a search with every character typed.  
**Solution:** Code fix.                          | QM001796447         |
| ADDM-14669 | **Problem:** Accept Windows smart quotes in Search window.  
**Solution:** Code fix.                           | QM001785927         |
| ADDM-14977 | **Problem:** CSV file export doesn't work if filename has special characters in it.  
**Solution:** Code fix.                          | QM001806401         |
| ADDM-15430 | **Problem:** Documentation does not give enough details on creating custom certificates.  
**Solution:** Code and Documentation fix as part of the Secure deployment feature. | QM001811351         |
| ADDM-15433 | **Problem:** Ugly error message on JDBC page if vault is closed.  
**Solution:** Code fix.                          | QM001852644, ISS04331364 |
| ADDM-15437 | **Problem:** AIX discovery may not fetch model and serial attributes preventing container to be created.  
**Solution:** Code fix.                          | QM001821312         |
| ADDM-15472 | **Problem:** Cluster exception selecting "Must Change Password" on deactivated account.  
**Solution:** Code fix.                           | QM001827908         |
| ADDM-15496 | **Problem:** Baseline check for DDD Removal Blackout Windows fails.  
**Solution:** Code fix.                            | QM001827921         |
| ADDM-15519 | **Problem:** Remove bogus serial number from virtinfo command in Solaris platforms.  
**Solution:** Code fix.                            | QM001829295         |
| ADDM-15533 | **Problem:** Proxy manager fails to create new proxies when BMC Atrium Discovery has a comma in the name.  
**Solution:** Code fix.                            | QM001829682         |
| ADDM-15554 | **Problem:** Adding new candidate for path to prtdiag command in Solaris platform script.  
**Solution:** Code fix.                            | QM001848221         |
| ADDM-15560 | **Problem:** ADDM fills the WWPN/WWNN fields inversely on AIX HBA.  
**Solution:** Code fix.                            | QM001824125         |
| ADDM-15576 | **Problem:** pdc (Primary Domain Controller) attribute should be removed from taxonomy.  
**Solution:** Code fix.                            | QM001830315         |
| ADDM-15584 | **Problem:** Documentation must specify that prior to updating the CMDB Sync configuration, Discovery and continuous CMDB synchronization must be stopped.  
**Solution:** Documentation fix.      | QM001823912         |
| ADDM-15588 | **Problem:** Windows standalone scanner should not wait for key if --quiet.  
**Solution:** Code fix.                            | QM001851323         |
| ADDM-15595 | **Problem:** HBA discovery parsing failure for ESX with Cisco and HP Hypervisors.  
**Solution:** Code fix.                            | QM001849943         |
| ADDM-15596 | **Problem:** Line Return Characters in the report of VIO (AIX) Discovered OS attribute.  
**Solution:** Code fix.                            | QM001849471         |
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-15622</td>
<td>Problem: aix_manufacturer_code not copied from DiscoveredHBA to inferred FibreChannelHBA. Solution: Code fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-15639</td>
<td>Problem: Coordinator fails to restart in the middle of a fixing rebalance. Solution: Code fix.</td>
<td>ISS04340164</td>
</tr>
<tr>
<td>ADDM-15644</td>
<td>Problem: control+alt+delete controlled through .override file. Solution: Code fix.</td>
<td>QM001857046</td>
</tr>
<tr>
<td>ADDM-15651</td>
<td>Problem: If you refine a search which explodes an already-exploded nodeset, it may cause a segmentation fault. Solution: Code fix.</td>
<td>QM001863832</td>
</tr>
<tr>
<td>ADDM-15655</td>
<td>Problem: Problem with tw_report_usage in ADDM 10.0.00. Solution: Code fix.</td>
<td>QM001857106</td>
</tr>
<tr>
<td>ADDM-15714</td>
<td>Problem: Documentation must be clear about the Component Filter tab. Solution: Documentation fix.</td>
<td>QM001824325</td>
</tr>
</tbody>
</table>

**Defects Resolved in BMC Atrium Discovery Version 10.0 Patch 2**

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-15487</td>
<td>Problem: AUTOSTART_SCANS option does not take effect. Discovery scan status is not preserved across a restart. Resolution: Fixed in the code. This no longer occurs.</td>
<td>QM001827604</td>
</tr>
<tr>
<td>ADDM-15485</td>
<td>Problem: VMware vCenter discovery and SQL discovery integration results are not consolidated. Resolution: Fixed in the code. VMware vCenter discovery and SQL discovery integration results are now consolidated.</td>
<td>QM001824040</td>
</tr>
<tr>
<td>ADDM-15479</td>
<td>Problem: Collaborative Application Mapping observed communication function generates invalid query. Resolution: Fixed in the code. This no longer occurs.</td>
<td>QM001824296</td>
</tr>
<tr>
<td>ADDM-15478</td>
<td>Problem: Use of static DIP only on the consolidation appliance fails. Resolution: Fixed in the code. This no longer occurs.</td>
<td>QM001827566</td>
</tr>
</tbody>
</table>

**Defects Resolved in BMC Atrium Discovery Version 10**

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-15225</td>
<td>Problem: HTTP header limits added by 9.0 SP2 (STIG) are too low for some SSO/BMC Atrium Discovery integrations. Resolution: Code fix, this no longer occurs.</td>
<td>QM001817771</td>
</tr>
<tr>
<td>ADDM-15223</td>
<td>Problem: Model segfault after &quot;Error: a servant has been deleted that is still activated.&quot; Resolution: Code fix, this no longer occurs.</td>
<td>QM001818236</td>
</tr>
<tr>
<td>ADDM-15221</td>
<td>Problem: The permission to import CAM definition is &quot;admin/import/csv&quot;. Resolution: Code fix, this no longer occurs.</td>
<td>QM001816403</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
</tbody>
</table>
| ADDM-15210 | **Problem:** BMC Atrium Discovery is not able to discover WPAR on LPAR.  
**Resolution:** Code fix, this no longer occurs.                                                                                 | QM001817308  |
| ADDM-15155 | **Problem:** In BMC Atrium Discovery 9.0.1 the DiscoveredNetworkInterface node contains driver_version and driver_date on Windows discovery. However, the taxonomy does not contain them.  
**Resolution:** Code fix, this no longer occurs.                                                                 | QM001810892  |
| ADDM-15137 | **Problem:** BMC Atrium Discovery displays incorrect message when there are multiple CECs discovered using z/OS Agent 1.7.00.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001815058  |
| ADDM-15125 | **Problem:** BMC Atrium Discovery is not correctly parse the information about registry entries when the last part of key is repeated.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001814367  |
| ADDM-15102 | **Problem:** The tw_backup overwrite documentation is inconsistent.  
**Resolution:** Documentation fix.                                                                                                    | QM001812544  |
| ADDM-15081 | **Problem:** Discovery of OpenVMS 6.2 servers failed in getMACAddresses with "Unexpected output format".  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001810902  |
| ADDM-15080 | **Problem:** BAI name is not populated in PDF group report.  
**Resolution:** Code fix, this no longer occurs.                                                                                     | QM001810876  |
| ADDM-15074 | **Problem:** PDC attribute is only discoverable via remquery not WMI, but WMI is the primary access method.  
**Resolution:** Code fix, the PDC attribute is no longer set by BMC Atrium Discovery.                                                 | QM001810707  |
| ADDM-15070 | **Problem:** Documentation required for devices responding when nothing is connected?  
**Resolution:** Documentation fix.                                                                                                   | QM001810489  |
| ADDM-15018 | **Problem:** Scan of Solaris 11.1 does not pick up all NICs.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001808277  |
| ADDM-15012 | **Problem:** Solaris getInterfaceList fails with "unknown ifconfig output", igb interface.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001823805  |
| ADDM-15011 | **Problem:** Missing serial number in AIX machines when prtcconf and lscfg does not return it.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001808070  |
| ADDM-15003 | **Problem:** hostinfo method script is incorrectly looking for the prtdiag path for Solaris 11.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001807600  |
| ADDM-14997 | **Problem:** Cannot enable HTTPS when FIPS is enabled.  
**Resolution:** Code fix, this no longer occurs.                                                                                     | QM001809617  |
| ADDM-14974 | **Problem:** vCenter discovery: getVirtualMachines fails with exception when scanning one of the ESX 5 hypervisors.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001806741  |
| ADDM-14953 | **Problem:** Some SUN hosts do not have hardware reference even though model/vendor exists in Hardware Reference Data - Extended Data Pack 2013-Sep-1.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001806239  |
| ADDM-14947 | **Problem:** Hourly Average Traffic Statistics unable to display graphic for servers with more than three Ethernet adapters.  
**Resolution:** Code fix, this no longer occurs.                                                                                   | QM001805761  |
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-14920 | **Problem:** Windows credential deployment documentation provides incorrect download link.  
**Resolution:** Documentation fix. | QM001801178   |
| ADDM-14916 | **Problem:** File not readable by tideway causes disk configuration to fail to move log files (for example netadmin.log).  
**Resolution:** Code fix, this no longer occurs. | QM001804027   |
| ADDM-14913 | **Problem:** VMware tools install causes disk management to fail.  
**Resolution:** Code fix, this no longer occurs. | QM001804027   |
| ADDM-14900 | **Problem:** getIPAddresses() method failing for Lexmark printer.  
**Resolution:** Code fix, this no longer occurs. | QM001801258   |
| ADDM-14847 | **Problem:** Appliance Support page not accessible when special characters entered in the description field of the archive.  
**Resolution:** Code fix, this no longer occurs. | QM001799659   |
| ADDM-14837 | **Problem:** The "rescan now" button does not keep the company settings.  
**Resolution:** Code fix, this no longer occurs. | QM001798835   |
| ADDM-14835 | **Problem:** Documentation does not state VLAN information is no longer supported.  
**Resolution:** Documentation fix. | QM001798640   |
| ADDM-14832 | **Problem:** Documentation update request for VMware Tools installation procedure.  
**Resolution:** Documentation fix. | QM001771901   |
| ADDM-14822 | **Problem:** Duplicate CIs with the same ADDMIntegrationID.  
**Resolution:** Code fix, this no longer occurs. | QM001798190   |
| ADDM-14821 | **Problem:** OpenVMS discovery problem with different prompts.  
**Resolution:** Code fix, this no longer occurs. | QM001795133   |
| ADDM-14800 | **Problem:** The documentation for tw_scan_control doesn't mention the common options such as --password.  
**Resolution:** Documentation fix. | QM001795407   |
| ADDM-14799 | **Problem:** Too many files open omniORB.TRANSIENT_ConnectFailed  
**Resolution:** Code fix, this no longer occurs. | QM001798067   |
| ADDM-14798 | **Problem:** CMDB Connection Time Out.  
**Resolution:** Code fix, this no longer occurs. | QM001796870   |
| ADDM-14775 | **Problem:** DiscoveryAccess endtime is not in a readable format when displayed in list  
**Resolution:** Code fix, this no longer occurs. | QM001795579   |
| ADDM-14765 | **Problem:** BMC Atrium Discovery 8.0 documentation should include note for old versions of HMC.  
**Resolution:** Documentation fix. | QM001773395   |
| ADDM-14751 | **Problem:** UI can fail to correctly update mapping set baseline.  
**Resolution:** Code fix, this no longer occurs. | QM001794338   |
| ADDM-14744 | **Problem:** Cascade removal deletes BAI when required FunctionalComponent has some required content removed.  
**Resolution:** Code fix, this no longer occurs. | QM001794480   |
| ADDM-14740 | **Problem:** Documentation change requests related to Disk Configuration utility.  
**Resolution:** Documentation fix. | QM001774741   |
<p>|          |                                                                        | QM001792079   |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-14727</td>
<td>Problem: Missing description for a specific Discovery Configuration parameter (Maximum search window size).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14716</td>
<td>Problem: Incorrect RAM value for VMs.</td>
<td>QM001786554/ QM001788196</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14707</td>
<td>Problem: Functional component definition block UI screen not allowing interaction with software instances.</td>
<td>QM001793106</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14701</td>
<td>Problem: Unable to discover SCO_SV 5.0.7 servers - ifconfig not in path.</td>
<td>QM001791449</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14699</td>
<td>Problem: Mismatch on the discovered HP-UX RAM value.</td>
<td>QM001790047</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14696</td>
<td>Problem: BMC Atrium Discovery is not able to parse the new getMACaddress format in Tru64 Servers.</td>
<td>QM001790807</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14672</td>
<td>Problem: Could not display the data returned by filtering the OS in the Host list page when &quot;Others&quot; or certain OS is selected. It directs to The Technical error page.</td>
<td>QM001791015</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14664</td>
<td>Problem: DiscoveredNetworkConnection should be committed in batches to reduce memory usage in reasoning.</td>
<td>QM001790378</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14655</td>
<td>Problem: BMC Atrium Discovery is assigning incorrect subnet/netmask to ESXi host.</td>
<td>QM001786536</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14623</td>
<td>Problem: BMC Atrium Discovery 9.0 and 9.0.1 kickstart install fails on HP hardware.</td>
<td>QM001789134</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14621</td>
<td>Problem: Unexpected exception from Windows Credential Proxy when unicode &quot;§&quot; charcter is used in userid/password.</td>
<td>QM001788391</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14608</td>
<td>Problem: OpenVMS not accepting password prompt.</td>
<td>QM001788403</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14588</td>
<td>Problem: Unroutable targets not recognized as darkspace.</td>
<td>QM001786373</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14587</td>
<td>Problem: Error starting reasoning - consolidation config corruption.</td>
<td>QM001784763</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14585</td>
<td>Problem: Upgrade option &quot;--backup-overwrite&quot; is unrecognized</td>
<td>QM001785997</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14563</td>
<td>Problem: HP-UX machinfo contains: &quot;17 cores (1 to 4 per socket)&quot;. We don't understand a range (1 to 4) for cores_per_processor attribute.</td>
<td>QM001785430</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14556</td>
<td>Problem: I18N - A scheduled discovery run label with multi-bytes chars is not correctly displayed when editing.</td>
<td>QM001783611</td>
</tr>
<tr>
<td></td>
<td>Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>
| ADDM-14548 | **Problem:** The PDF reports generated and store in ~/var/reports are not captured in snapshots or v9 backup.  
**Resolution:** Code fix, this no longer occurs.                                                                                          | QM001784225     |
| ADDM-14550 | **Problem:** Query Builder doesn’t handle attributes names that are keywords.  
**Resolution:** Code fix, this no longer occurs.                                                                                           | QM001783121     |
| ADDM-14518 | **Problem:** UI changes to DNS are allowed even if DHCP is enabled, so they are lost at network restart.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001783112     |
| ADDM-14517 | **Problem:** ADDM: BOGUS_SERIAL_FILTER should include "To Be Filled By O.E.M.".  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001780173     |
| ADDM-14506 | **Problem:** BMC Atrium Discovery incorrectly classifying dark space as skipped instead of no response.  
**Resolution:** Code fix, this no longer occurs.                                                                                           | QM001783344     |
| ADDM-14505 | **Problem:** Scanners do not automatically select the default company as set on the consolidator when adding a discovery run.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001775976     |
| ADDM-14494 | **Problem:** For listDirectory in 8.3 and 9: PRIV_TEST and PRIV_SUDO do not work and there is inadequate error checking.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001781179     |
| ADDM-14493 | **Problem:** WMI queries fail for providers root\ADPT, root\IBMSD, and root\PG_Interop.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001781110     |
| ADDM-14487 | **Problem:** getInterfaceList returns status of "UNKNOWN_PythonException" for some HP-UX hosts  
**Resolution:** Code fix, this no longer occurs.                                                                                          | QM001777420     |
| ADDM-14469 | **Problem:** CMDB device filter evaluation is incorrect when more than one "None" conditional conjunction.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001778221     |
| ADDM-14465 | **Problem:** MAC to IP Report data is truncated when exported to CSV file.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001779011     |
| ADDM-14460 | **Problem:** Missing CPU information for HP-UX host: machinfo: Unexpected output format.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001778972     |
| ADDM-14448 | **Problem:** When mapping the application in CAM, it is possible to generate TPL that won't compile.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001779709     |
| ADDM-14433 | **Problem:** Tighten permissions on /etc/rc.d/init.d/omniNames to -rwxr-xr-x  
**Resolution:** Code fix.                                                                                                                | QM001777487     |
| ADDM-14424 | **Problem:** Unable to download CAM generated PDF report if the group name ends with "+".  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001768968     |
| ADDM-14412 | **Problem:** The company name setting is not i18N compliant.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001773447     |
| ADDM-14411 | **Problem:** HRD Import disclose the location of the software in case of an error.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001772404     |
<p>| ADDM-14406 |                                                                                                                                                                                                     | QM001776724     |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-14405 | **Problem:** Some Solaris zombie processes can return no cmd name in DiscoveredProcesses; results in ECA error.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                 | QM001776558   |
| ADDM-14388 | **Problem:** Linux (and other?) zombie processes should have a process cmd name that does not trigger SI creation.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001775417   |
| ADDM-14385 | **Problem:** When a cmdb device filter uses a DDD node, sync stops deleting destroyed CIs.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                         | QM001775182   |
| ADDM-14378 | **Problem:** Consolidation of static DIP requests does not work.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001775865   |
| ADDM-14361 | **Problem:** Discovery runCommand does not handle tw_report output.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           |               |
| ADDM-14337 | **Problem:** For AIX, the 'getHostInfo' script isn't picking up serial number. The session log shows: Machine Serial Number: Not Available.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001772011   |
| ADDM-14336 | **Problem:** Cannot discover virtual AIX HBA cards.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                               | ISS03934194/QM001750559 |
| ADDM-14293 | **Problem:** SI belonging to another node gets marked as deleted in the CMDB.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                         | QM001768414   |
| ADDM-14226 | **Problem:** When loading a pattern, "|..." is being replaced in the display by ") a".  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                             | QM001766429   |
| ADDM-14188 | **Problem:** Inconsistent size limitation for Group name.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                             | QM001767562   |
| ADDM-14141 | **Problem:** Proxy test credential fails with "cannot access" error if the IP address does not match the range of the pool.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001761754   |
| ADDM-14075 | **Problem:** Error parsing getDirectoryListing results when ACL is defined.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                         | QM001759781   |
| ADDM-14068 | **Problem:** Reported errors with the tw_health_check command.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001759255   |
| ADDM-13992 | **Problem:** Discovery change at 8.3.02 means some Solaris OSIs now get collapsed to one host node due to use of sneep's serial number.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                         | QM001753496   |
| ADDM-13954 | **Problem:** Wrong OS string data for Solaris when patch bundle applied.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001751012   |
| ADDM-13381 | **Problem:** CE VM shows 0 Actual memory in the Appliance specification page.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                         | QM001784426   |
| ADDM-13369 | **Problem:** NullPointerException discovering WebLogic over JMX.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                                                                                                                                                                                                                                           | QM001753683   |
### Defects Resolved in BMC Atrium Discovery Version 9.0 SP2

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-13300</td>
<td><strong>Problem:</strong> Occasional ECA errors discovering IP address via SNMP.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001754489</td>
</tr>
<tr>
<td>ADDM-12852</td>
<td><strong>Problem:</strong> Scans failed with &quot;Error (Unable to get the deviceInfo: User cancelled request)&quot;.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001693700</td>
</tr>
<tr>
<td>ADDM-12308</td>
<td><strong>Problem:</strong> Cannot generate a bar graph for a scan containing an apostrophe.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001677770</td>
</tr>
<tr>
<td>ADDM-11648</td>
<td><strong>Problem:</strong> Unusable system after deleting TKU.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001650005</td>
</tr>
<tr>
<td>ADDM-9782</td>
<td><strong>Problem:</strong> Pattern module may be silently disabled when another module with a clashing namespace is loaded.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001738960</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-15089</td>
<td><strong>Problem:</strong> When scanning Oracle Solaris 11.1, BMC Atrium Discovery fails to parse network interface card (NIC) names which contain dots (.). For example, BMC Atrium Discovery will fail to parse a NIC name like net9.2054.  &lt;br&gt; <strong>Resolution:</strong> Fixed in the code.</td>
<td>QM001808277</td>
</tr>
<tr>
<td>ADDM-14958</td>
<td><strong>Problem:</strong> A memory management flaw in the search functionality of BMC Atrium Discovery leads to gradual memory consumption increase of the model process.  &lt;br&gt; <strong>Resolution:</strong> Fixed in the code.</td>
<td>QM001781110</td>
</tr>
<tr>
<td>ADDM-15092</td>
<td><strong>Problem:</strong> When a new patch is released (that is, n.n.n.X), BMC Atrium Discovery considers this as a different version and prevents restoring backups made immediately before the patch was applied. &lt;br&gt; <strong>Resolution:</strong> Fixed in the code. Restore now accepts backups where the Service Pack version (n.n.n) matches and the patch version (n.n.n.X) is not later than the current level.</td>
<td>QM001785308</td>
</tr>
<tr>
<td>ADDM-14499</td>
<td><strong>Problem:</strong> On 64 bit Windows systems, the WMI queries to the 32 bit providers fail and report Provider load failure.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001781110</td>
</tr>
<tr>
<td>ADDM-14553</td>
<td><strong>Problem:</strong> On Linux, based on whether ifconfig or ip is used, the getMACAddresses and getNetworkInterfaces scripts return different MAC addresses and interfaces.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001785997</td>
</tr>
<tr>
<td>ADDM-14554</td>
<td><strong>Problem:</strong> On Linux, the getNetworkInterfaces script requires ip to be found in /sbin/ip.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001785308</td>
</tr>
<tr>
<td>ADDM-14555</td>
<td><strong>Problem:</strong> Parsing of the iproute2 output for the newer versions of iproute2 (for example, the version in Fedora 18) fails.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001785308</td>
</tr>
<tr>
<td>ADDM-14557</td>
<td><strong>Problem:</strong> When you upgrade to version 9.0 SP1 on RHEL 6, the upgrade script deletes the existing firewall (iptables).  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001785308</td>
</tr>
<tr>
<td>ADDM-14579</td>
<td><strong>Problem:</strong> The getMACAddress method fails on HP-UX versions earlier than 11.0.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001785997</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>ADDM-14584</td>
<td>Problem: The upgrade option, --backup--overwrite, is not recognized. Resolution: Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14590</td>
<td>Problem: When you scan darkspaces, BMC Atrium Discovery 9.0 may report the discovery error Unable to get the deviceInfo: Execution Failure instead of NoResponse. Resolution: Code fix. this no longer occurs.</td>
<td>QM001786373</td>
</tr>
<tr>
<td>ADDM-14619</td>
<td>Problem: If HTTPS is enabled, it is possible to disable FIPS mode. Resolution: Code fix, FIPS mode can no longer be disabled.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14620</td>
<td>Problem: Discovery of OpenVMS can fail to recognize the password prompt causing BMC Atrium Discovery to report NoAccess even with a valid password. Resolution: Code fix, this no longer occurs.</td>
<td>QM001788403</td>
</tr>
<tr>
<td>ADDM-14628</td>
<td>Problem: The BMC Atrium Discovery CMDB synchronization mechanism does not handle slicing lists and strings. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14629</td>
<td>Problem: BMC Atrium Discovery versions 9.0 and 9.0 SP1 kickstart installation fails on HP ProLiant BL460c hardware. Resolution: Code fix, this no longer occurs.</td>
<td>QM001789134</td>
</tr>
<tr>
<td>ADDM-14645</td>
<td>Problem: The Host List page fails to display those discovered operating systems for which the Discovered OS filter column contains values with non-ASCII characters. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14656</td>
<td>Problem: For VMware ESXi hosts, BMC Atrium Discovery may assign incorrect subnet and netmask. Resolution: Code fix, this no longer occurs.</td>
<td>QM001786536</td>
</tr>
<tr>
<td>ADDM-14657</td>
<td>Problem: Searching for a host with a fully qualified domain name (FQDN) as the search expression may fail to return results. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14663</td>
<td>Problem: On AIX, discovering the exact number of cores on the newer versions of IBM POWER microprocessors is not possible. Resolution: Code fix, discovering the number of IBM POWER microprocessors is no longer supported.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14665</td>
<td>Problem: The memory requirement for appliances with large number of network connections is high. Resolution: Code fix by improving caching efficiency and batching large data chunks.</td>
<td>QM001790378</td>
</tr>
<tr>
<td>ADDM-14674</td>
<td>Problem: On Internet Explorer, when you access the Host List page and filter the discovered operating systems by selecting Others from the Filter Column, the values containing non-ASCII characters are not displayed. Resolution: Code fix, this no longer occurs.</td>
<td>QM001791015</td>
</tr>
<tr>
<td>ADDM-14681</td>
<td>Problem: During an upgrade, an invalid warning about a pattern using a deprecated attribute may be displayed. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>ADDM-14682</td>
<td>Problem: CPU and memory information is not displayed for some discovered HP-UX hosts. Resolution: Code fix, this no longer occurs.</td>
<td>QM001778972</td>
</tr>
<tr>
<td>ADDM-14684</td>
<td>Problem: When you edit a scheduled discovery run, the fields on the Edit an Existing Run dialog are not correctly initialized. Resolution: Code fix, this no longer occurs.</td>
<td>QM001792097</td>
</tr>
</tbody>
</table>

**BMC Atrium Discovery 10.2**
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-14688 | **Problem:** Excessive memory usage on the consolidation and scanning appliances.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001793106   |
| ADDM-14708 | **Problem:** The query builder (which is also used in collaborative application mapping) fails to list any attributes if the node kind contains attributes name differing only in case (for example, *port* and *Port*).  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |
| ADDM-14711 | **Problem:** The disk configuration utility does not consider the system data and possible core dumps while estimating the size of record data.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |
| ADDM-14712 | **Problem:** The `getMACaddress` method may fail to parse Tru64 NetRAIN information.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001790807   |
| ADDM-14741 | **Problem:** For Software Instances, when you select **Detailed Observed Communication** from the **Reports** drop down, the technical difficulties page is displayed.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |
| ADDM-14745 | **Problem:** When you delete some of the contents of a required Functional Component, the Functional Component itself is deleted and in turn causes the BAI to be deleted.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001794480   |
| ADDM-14755 | **Problem:** The appliance's UI may fail to correctly display the updated mapping set baseline.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001794338   |
| ADDM-14768 | **Problem:** When you perform a system backup, it allows certain special characters in the notes field. However, when you attempt to restore, it fails until the special characters are removed.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |
| ADDM-14813 | **Problem:** When BMC Atrium Discovery is installed, HTTPS is enabled by default.  
**Resolution:** Code fix, HTTPS is no longer enabled by default.                                                                                     |                |
| ADDM-14824 | **Problem:** Reports and queries producing list of times fail to convert the time into a readable format.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001795579   |
| ADDM-14827 | **Problem:** When mapping an application using Collaborative Application Mapping (CAM), it is possible to generate a TPL that will not compile.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |

In addition to the defects in the preceding table, version 9.0 SP2 has a number of security enhancements and QM001789294, QM001803416, QM001806008, and QM001789294 has also been fixed.

### Defects Resolved in BMC Atrium Discovery Version 9.0 SP1

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| ADDM-14508 | **Problem:** BMC Atrium Discovery incorrectly classifies dark space as skipped instead of no response.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     | QM001783344   |
| ADDM-14491 | **Problem:** The estimated time displayed by on the UI for backing up and restoring the appliance can be extremely inaccurate.  
**Resolution:** Code fix, this no longer occurs.                                                                                                     |                |
<p>| ADDM-14475 |                                                                                                                                                                                                            |                |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Code</th>
<th>Description</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-14471</td>
<td></td>
<td><strong>Problem:</strong> The ECA engine might detect an error message for the network devices with no network interfaces. RuleError on rule edge_connectivity__validate_network_links due to: Error while executing a rule - TypeError: 'NoneType' object is not iterable.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14470</td>
<td></td>
<td><strong>Problem:</strong> If a large number of different endpoints have been scanned, viewing the SNMP Recognition Rules page can cause the appliance to report that it has no devices.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14466</td>
<td></td>
<td><strong>Problem:</strong> BMC Atrium Discovery 8.3 SP2 uses the sneep command to retrieve serial numbers on Solaris systems. For some Solaris global zones, the same serial number is returned by sneep. To merge hosts in these systems, BMC Atrium Discovery uses partition_id as a criteria, which is set by combining the non-unique serial number with the non-unique global zone. As a result, these systems collapse into one.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14464</td>
<td></td>
<td><strong>Problem:</strong> The results from EXPLODE queries exported to CSV file may be incomplete.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14453</td>
<td></td>
<td><strong>Problem:</strong> CPU information is not displayed for the discovered some HPUX hosts.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14428</td>
<td></td>
<td><strong>Problem:</strong> For defunct processes on Linux, Solaris, HPUX, and AIX, patterns may create Software Instance (SI) nodes.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14414</td>
<td></td>
<td><strong>Problem:</strong> Solaris defunct processes may return no value for DiscoveredProcess cmd which results in ECA error.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14402</td>
<td></td>
<td><strong>Problem:</strong> When a BMC Atrium CMDB device filter uses a DDD node, synchronization stops deleting the destroyed CIs.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14399</td>
<td></td>
<td><strong>Problem:</strong> While scanning Solaris hosts, Discovery Conditions may fail with an ECA error, RuleError on rule tpl_defn_functions_fn_getDiscoveryUserID_body_0 due to: Error while executing a rule - ValueError: invalid literal for int() with base 10.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14379</td>
<td></td>
<td><strong>Problem:</strong> Static data integration point (DIP) requests are not consolidated and report an error message, Request for information not part of the consolidated data.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14349</td>
<td></td>
<td><strong>Problem:</strong> When you export a search result which contains nodeLink, incorrect output is produced.</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ADDM-14346</td>
<td></td>
<td><strong>Problem:</strong> When pasting a list of IP addresses on the pill UI, invalid pills are not placed first (unless sorted by type).</td>
<td>Code fix, this no longer occurs.</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>ADDM-14345</td>
<td>Problem: While adding a new run from the Add New Run dialog, it was possible to click OK even if invalid IP address ranges were entered, which subsequently removes all the IP address ranges entered for the run. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14343</td>
<td>Problem: When the BMC Atrium Discovery log files are moved to a different disk using the disk configuration utility, the <code>tw_compress_logs</code> script run by cron does not compress the old logs and the <code>-tideway/log</code> symlink is automatically deleted 30 days after the initial log file was created. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14327</td>
<td>Problem: In Solaris hosts, failure to parse kstat CPU information causes failure of host discovery. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14322</td>
<td>Problem: If the omniNames service has been stopped, backing up the appliance fails and running the <code>tw_backup</code> command reports the error message:`Backup create normally runs when appliance services are running, either start services or use --force option. Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14318</td>
<td>Problem: Even though the IPv6 site local addresses are not scanned by BMC Atrium Discovery, the pill UI does not show the IPv6 site local addresses as invalid. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14289</td>
<td>Problem: When adding an IP address to a scheduled range, pressing return to convert to a pill works, but entering a second return dismisses the dialog without adding the additional IP address. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14274</td>
<td>Problem: In BMC Atrium Discovery version 9.0, when you click on a discovered host which has multiple network interfaces that do not have MAC addresses, the UI fails with a traceback. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14265</td>
<td>Problem: While migrating from version 8.3.x to 9.0, if services are not running, the <code>tw_addm_8_3_migration</code> script gives a traceback instead of an error message. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14230</td>
<td>Problem: Where hosts are related through a BAI, in some cases a sync of one host can cause SIs on another related hosts to be marked as deleted. Resolution: Code fix, this no longer occurs.</td>
<td>QM001768414</td>
<td></td>
</tr>
<tr>
<td>ADDM-14039</td>
<td>Problem: Single sign on does not support extracting values from X.509 certificate extensions. Resolution: Support added.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-14038</td>
<td>Problem: Configuring certificate revocation lists (CRLs) for HTTPS is not supported. Resolution: Support added.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDM-13985</td>
<td>Problem: Cannot discover virtual AIX HBA cards. Resolution: Code fix, this no longer occurs.</td>
<td>QM001750559</td>
<td></td>
</tr>
<tr>
<td>ADDM-14286</td>
<td>Problem: SNMP discovery of AIX VIO incorrectly displays the Operating System as AIX, instead of VIO. Resolution: Code fix, this no longer occurs.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In addition to the defects in the preceding table, version 9.0 SP1 has a number of security enhancements and QM001789294, QM001803416, QM001806008, QM001753496, QM001772404, and QM001777487 have also been fixed.
# Defects Resolved in BMC Atrium Discovery Version 9.0

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
</table>
| 10738| **Problem:** Search cannot match non-UK characters, for example characters with an umlaut (ğ).  
**Resolution:** Code fix, this no longer occurs.                                                                                           | 00013640      |
| 12700| **Problem:** The `PROCESSWITH` ordering in the search and reporting service is unreliable.  
**Resolution:** Code fix, this no longer occurs.                                                                                           |               |
| 14245| **Problem:** CMDB synchronization error when starting services after running `tw_model_init`.  
**Resolution:** Code fix, this no longer occurs.                                                                                           | QM001703499   |
| 14365| **Problem:** Discovery start time shown on consolidation server is off by five hours when both Discovery and Consolidation Servers are localized.  
**Resolution:** Code fix, this no longer occurs.                                                                                           | QM001692983   |
| 15035| **Problem:** Automounter is enabled in the appliance.  
**Resolution:** Code fix, this no longer occurs.                                                                                           | QM001703823   |
| 15175| **Problem:** The `tw_vmware` script is obsolete and should be removed.  
**Resolution:** Code fix, this has been removed.                                                                                         | QM001724848   |
| 15197| **Problem:** The "default" entry in PrimaryCapability_CTI_Mapping Table is not used when no capability matches.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001725494   |
| 15407| **Problem:** UNIX `getFileSystem` discovery scripts not catering for protected filesystems.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001724526   |
| 15423| **Problem:** "Lock table is out of available lock entries" (in the model), while trying to deactivate patterns.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001732061   |
| 15574| **Problem:** MDB test connection failed when the AR server was encrypted using BMC Remedy Encryption Security Product.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001737346   |
| 15660| **Problem:** Daily disk usage statistics graph fails to load data when long filesystem names are used.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001738802   |
| 15688| **Problem:** Manual pattern execution does not work for Network Devices.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001739731   |
| 15698| **Problem:** Read-only users should not be allowed attach documents to host nodes.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001738408   |
| 15713| **Problem:** Some Fibre Channel HBA attributes are missing after scanning VMWare ESXi 4.1.0/4.0.0/5. 0.0.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001740226   |
| 15719| **Problem:** The `partition_id` attribute is not populated on AIX & Linux on POWER LPARs.  
**Resolution:** Code fix, this no longer occurs.                                                                                         | QM001741255   |
<p>| 15727|                                                                                                                                           | QM001741327   |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>15739</td>
<td>Problem: Appliance Specification UI incorrectly reports disk space if the path includes a symlink. Resolution: Code fix, this no longer occurs.</td>
<td>QM001742085</td>
</tr>
<tr>
<td>15741</td>
<td>Problem: Solaris returns inconsistent model names for same O/S version. Resolution: Code fix, this no longer occurs.</td>
<td>QM001736818</td>
</tr>
<tr>
<td>15810</td>
<td>Problem: Unused permissions still available. Resolution: Code fix, permissions removed.</td>
<td>QM001750214</td>
</tr>
<tr>
<td>15815</td>
<td>Problem: PDF Reports with notes fail with an exception in docutils. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03921246</td>
</tr>
<tr>
<td>15840</td>
<td>Problem: The datastore (Model Maintenance -&gt; Size Warning) has a maximum of 100GB. This is too small. Resolution: Code fix, this has been changed.</td>
<td>QM001731647</td>
</tr>
<tr>
<td>15866</td>
<td>Problem: Processor speed is not obtained from most HP-UX servers. Resolution: No acceptable fix available, resolving as won't fix.</td>
<td>QM001745735, QM001753527</td>
</tr>
<tr>
<td>15938</td>
<td>Problem: Component Filter UI sometimes ignores present filter condition's containers. Resolution: Code fix, this no longer occurs.</td>
<td>QM001748972</td>
</tr>
<tr>
<td>15941</td>
<td>Problem: Discovery process fails on servers with Open Unix 8.0.0 OS. Resolution: Code fix, this no longer occurs.</td>
<td>QM001746127</td>
</tr>
<tr>
<td>15958</td>
<td>Problem: Failure to capture all of the output from the iterated <code>lscfg -vl $adapter 2&gt;/dev/null</code> command on AIX. Resolution: Code fix, this no longer occurs.</td>
<td>QM001741975</td>
</tr>
<tr>
<td>15964</td>
<td>Problem: Search service fails to expand explode in nodeset before applying filter. Resolution: Code fix, this no longer occurs.</td>
<td>QM001749389</td>
</tr>
<tr>
<td>15976</td>
<td>Problem: VMware ESX is incorrectly identified as Red Hat Linux. Resolution: Code fix, this no longer occurs.</td>
<td>QM001752148</td>
</tr>
<tr>
<td>15978</td>
<td>Problem: AIX/filesystem parsing error when a partition is reported as -1% used. Resolution: Code fix, this no longer occurs.</td>
<td>QM001749821</td>
</tr>
<tr>
<td>15983</td>
<td>Problem: Ugly CORBA permission error in Change Password UI when users do not have the security/user/passwd permission. Resolution: Code fix, this no longer occurs.</td>
<td>QM001749017</td>
</tr>
<tr>
<td>15984</td>
<td>Problem: The <code>/var/log/secure</code> files fill with sudo nmap traces, which might fill <code>/var</code>. Resolution: Code fix, this no longer occurs.</td>
<td>QM001749172</td>
</tr>
<tr>
<td>15985</td>
<td>Problem: After rebooting the appliance, IP Range configuration in Windows Proxy pool is lost. Resolution: Code fix, this no longer occurs.</td>
<td>QM001748266</td>
</tr>
<tr>
<td>15986</td>
<td>Problem: SUSE Linux reports wrong versioning. Resolution: Code fix, this no longer occurs.</td>
<td>QM001748433</td>
</tr>
<tr>
<td>15988</td>
<td>Problem: Incorrect file name when saving a visualization as an image. Resolution: Code fix, this no longer occurs.</td>
<td>QM001743331</td>
</tr>
<tr>
<td>15992</td>
<td></td>
<td>QM001737602</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>15994</td>
<td>Problem: A different user name is displayed in the UI where more than one user has a similar first name.</td>
<td>QM001750127</td>
</tr>
<tr>
<td>15995</td>
<td>Problem: Group permission page needs more detail.</td>
<td>QM001729094,  QM001752806</td>
</tr>
<tr>
<td>16021</td>
<td>Problem: On first scan of an ESX host via ssh, discovery was incomplete, and the session log reported “Command not found” errors.</td>
<td>QM001754317</td>
</tr>
<tr>
<td>16028</td>
<td>Problem: Parser should handle the command failure gracefully when fcinfo failed to load FC-HBA common library.</td>
<td>QM001750209</td>
</tr>
<tr>
<td>16032</td>
<td>Problem: Solaris discovery script (DeviceInfo) contains –f file test condition, it should be –r.</td>
<td>QM001750210</td>
</tr>
<tr>
<td>16057</td>
<td>Problem: The IBM.DB2.RDBMS_MF_Extended.DB2_Databases pattern takes too long for large mainframe databases.</td>
<td>QM001751695</td>
</tr>
<tr>
<td>16092</td>
<td>Problem: Incorrect cores calculation on HP-UX.</td>
<td>QM001752617</td>
</tr>
<tr>
<td>16099</td>
<td>Problem: ECA error in traverseFindOrCreateNode for integer values too large in df data.</td>
<td>QM001749794</td>
</tr>
<tr>
<td>16103</td>
<td>Problem: Incorrect OS discovered for Oracle Linux &gt;= 5.3.</td>
<td>QM001749279</td>
</tr>
<tr>
<td>16124</td>
<td>Problem: Scan Failure on Linux when /etc/*-release file is empty or has no read permissions. OS is reported as (none).</td>
<td>QM001751719</td>
</tr>
<tr>
<td>16127</td>
<td>Problem: The OpenVMS processes are not parsed where they include a space character and/or where the process cmd is displayed on multiple lines.</td>
<td>QM001753180</td>
</tr>
<tr>
<td>16135</td>
<td>Problem: SPARC T3 processor type shown as Unknown (SPARC-T3).</td>
<td>QM001748489</td>
</tr>
<tr>
<td>16142</td>
<td>Problem: ESX blades report wrong serial number.</td>
<td>QM001753771</td>
</tr>
<tr>
<td>16146</td>
<td>Problem: The vim-cmd-interfaces output is parsed incorrectly.</td>
<td>QM001754028</td>
</tr>
<tr>
<td>16162</td>
<td>Problem: Login session failed because MAX_READ_SIZE is too big and the searched string could not be found within MAX_SEARCH_SIZE.</td>
<td>QM001754985</td>
</tr>
<tr>
<td>16173</td>
<td>Problem: Linux parser error when parsing hba_procsfs output in which BIOS, FCODE, EFI and Flash information does not exist.</td>
<td>QM001755689</td>
</tr>
<tr>
<td>16179</td>
<td>Problem: Add discovery for package vendor, description, and epoch for Linux, Solaris, and HP-UX.</td>
<td>QM001709962</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>16182</td>
<td><strong>Problem:</strong> Report generation sticks at 95%.</td>
<td>QM001755722</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16186</td>
<td><strong>Problem:</strong> Should not report general parser error when hba_sysfs contains data but returns no HBA data.</td>
<td>QM001755686</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16194</td>
<td><strong>Problem:</strong> Query Builder incorrectly displays/selects different non-taxonomy attributes with identical printable names.</td>
<td>QM001757280</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16203</td>
<td><strong>Problem:</strong> Solaris machines on which serial numbers are not discovered can collapse OSIs.</td>
<td>QM001754173</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16243</td>
<td><strong>Problem:</strong> Improper HBA Discovery on Windows 2008 Server SP2.</td>
<td>QM001755297</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16248</td>
<td><strong>Problem:</strong> You cannot discover SUSE 12.1 with a root credential.</td>
<td>QM001760902</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16255</td>
<td><strong>Problem:</strong> Duplicate Host nodes are created when a host has two IP addresses, one of which is “locked down”.</td>
<td>QM001759675</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs. The snapshot feature has been replaced with appliance backup and restore.</td>
<td>QM001760001</td>
</tr>
<tr>
<td>16267</td>
<td><strong>Problem:</strong> Synchronization into CMDB updates CIs that have not been modified.</td>
<td>QM001759740</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16342</td>
<td><strong>Problem:</strong> Win32_PhysicalMemory fails with Error ‘Capacity’ when the memory bank is empty.</td>
<td>QM001763233</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16385</td>
<td><strong>Problem:</strong> Tru64 discovery fails to discover RAM.</td>
<td>QM001765619</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16405</td>
<td><strong>Problem:</strong> The hwmgz command must be run with escalated privileges to get the model number on Tru64.</td>
<td>QM001766276</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001764951</td>
</tr>
<tr>
<td>16458</td>
<td><strong>Problem:</strong> If you create a Group node in TPL you need to provide __pdf_detail_level</td>
<td>QM001759466</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16464</td>
<td><strong>Problem:</strong> Improve documentation for the process for adding swap as a file or a new disk/partition.</td>
<td>QM001768697</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code and documentation fix. See Managing disks and swap space.</td>
<td>QM001789294, QM001803416, QM001806008, QM001761830, QM001751640, and QM001768697 are also fixed.</td>
</tr>
</tbody>
</table>

**Defects Resolved in BMC Atrium Discovery Version 8.3 SP3**

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>16605</td>
<td><strong>Problem:</strong> Win32_PhysicalMemory fails with Error ‘Capacity’ when the memory bank is empty.</td>
<td>QM001763233</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>16591</td>
<td><strong>Problem:</strong> Duplicate Host nodes are created when a host has two IP addresses, one of which is “locked down”.</td>
<td>QM001750559</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>16552</td>
<td><strong>Problem:</strong> Cannot discover virtual AIX HBA cards. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001768414</td>
</tr>
<tr>
<td>16423</td>
<td><strong>Problem:</strong> Where hosts are related through a BAI, in some cases a sync of one host can cause SIs on another related hosts to be marked as deleted. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001754173</td>
</tr>
<tr>
<td>16407</td>
<td><strong>Problem:</strong> The &quot;default&quot; entry in the PrimaryCapability_CTI_Mapping Table is not used when no capability matches. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001725494</td>
</tr>
<tr>
<td>16406</td>
<td><strong>Problem:</strong> ECA error in traverseFindOrCreateNode for integer values too large in 'df' data. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001749794</td>
</tr>
<tr>
<td>16399</td>
<td><strong>Problem:</strong> Synchronization into CMDB updates CIs that have not been modified. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001759740</td>
</tr>
<tr>
<td>16110</td>
<td><strong>Problem:</strong> Bug in JDK causing SQL Server connections to hang. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001752103</td>
</tr>
<tr>
<td>16089</td>
<td><strong>Problem:</strong> HMC/getInterfaceList fails with exception because there is no 'speed'. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001752103</td>
</tr>
<tr>
<td>15764</td>
<td><strong>Problem:</strong> Discovery dashboard channels using old DDD aging option. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001733236</td>
</tr>
<tr>
<td>15762</td>
<td><strong>Problem:</strong> The model does not start after running <code>tw_ds_compact -x</code>. The option has been removed from the utility and the documentation. <strong>Resolution:</strong> Code and documentation fix.</td>
<td>QM001733236</td>
</tr>
</tbody>
</table>

QM001803416, QM001806008, QM001805996, QM001772492 and QM001768697 are also fixed.

**Defects Resolved in BMC Atrium Discovery Version 8.3 SP2**

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>12534</td>
<td><strong>Problem:</strong> Serial number not obtained on some Solaris/SPARC machines. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001731710</td>
</tr>
<tr>
<td>12827</td>
<td><strong>Problem:</strong> Visualizations only print in hierarchical view. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001687517 QM001659086</td>
</tr>
<tr>
<td>13119</td>
<td><strong>Problem:</strong> Processor speed conversion error. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001687517 QM001659086</td>
</tr>
<tr>
<td>13705</td>
<td><strong>Problem:</strong> Authentication cookie is not HTTPOnly. <strong>Resolution:</strong> Fixed in the code.</td>
<td>QM001676012</td>
</tr>
<tr>
<td>13736</td>
<td><strong>Problem:</strong> Misleading error when using incorrect MS SQL Server JDBC Driver. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>QM001654421</td>
</tr>
<tr>
<td>13950</td>
<td><strong>Problem:</strong> Only the maximum speed is discovered for multiple processor hosts. <strong>Workaround:</strong> None.</td>
<td>QM001659086</td>
</tr>
<tr>
<td>ID</td>
<td>Details</td>
<td>Defect Number</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
</tbody>
</table>
| 14925| **Problem:** Fujitsu machine has vendor "Sun Microsystems".  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                      | QM001715432   |
| 14939| **Problem:** The visualization and PDF report fail when a host with a Japanese name is included.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                  |               |
| 14942| **Problem:** A failure during database discovery results in an IntegrationResult with a node as an attribute. The following error is displayed on the scanning appliance: "Unexpected exception while sending data to the consolidation appliance".  
**Resolution:** Code fix, this no longer occurs.                                                                                               | QM001719307   |
| 14951| **Problem:** Invalid raw_speed on PortInterfaces. On high speed interfaces such as 10Gb/s, ifHighSpeed is not read, instead the value for raw_speed is shown in PortInterfaces. This is usually 4294.967295 Mb/s or similar.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                   | QM001734820   |
| 15171| **Problem:** Mainframe discovery uses incorrect view area code for MQ Listeners.  
**Resolution:** Fixed in the code. See also Mainframe discovery commands.                                                                                       |               |
| 15207| **Problem:** TRANSIENT_CallTimedOut errors occur when SNMP discovery attempts to scan some very complex network devices.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                   |               |
| 15218| **Problem:** If you have viewed a chart from results obtained using the query builder, using the back button does not work correctly.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                  |               |
| 15235| **Problem:** You can only select and import one credential type at a time otherwise the import will fail silently and simply refresh the Credential Migration page.  
**Resolution:** Fixed in the code.                                                                                                                                                    |               |
| 15273| **Problem:** Consolidation start time is misleading.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                      | QM001722913   |
| 15283| **Problem:** Tripwire looking for BerkeleyDB.4.7, and incorrectly named image, background4.jpg should read background4.gif.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                  | QM001735294   |
| 15285| **Problem:** Cannot cancel consolidation of a run that has been canceled on the scanner.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                         | QM001722947   |
| 15288| **Problem:** Legacy Mainframe MDZ code used to generate XML is inefficient.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                  | QM001723073   |
| 15302| **Problem:** The OS classifier identifies all variants (Novell SUSE Linux Enterprise Server (9.0 - 11.0), SUSE Linux Enterprise Server (v11 onwards), and OpenSUSE) of SUSE Linux as 'SUSE Linux'.  
**Resolution:** Code fix, this no longer occurs. This fix is also included in TKU releases 2012-01-10 and later.  |               |
| 15327| **Problem:** Search get() function accepts NodeHandles but not nodes.  
**Resolution:** Fixed in the code.                                                                                                                                                    | QM001723410   |
| 15356| **Problem:** ECA engine is always O(n) on number of root trackers.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                  | QM001731315   |
| 15390| **Problem:** Daily SAR chart shows nonsense data, a date in the I/O.  
**Resolution:** Code fix, this no longer occurs.                                                                                                                                  | QM001730013   |
| 15400|                                                                                                                                                                                                                                                                                                                                        |               |
## Defects resolved in BMC Atrium Discovery version 8.3 SP1

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>15399</td>
<td><strong>Problem:</strong> Legacy Mainframe MDZ code used to generate XML is inefficient. <strong>Resolution:</strong> Fixed in the code.</td>
<td>QM001723073</td>
</tr>
<tr>
<td>15400</td>
<td><strong>Problem:</strong> A failure during Dynamic Dip database discovery results in an IntegrationResult with a node as an attribute. The following error is displayed on the scanning appliance: &quot;Unexpected exception while sending data to the consolidation appliance&quot;. <strong>Resolution:</strong> Fixed in the code.</td>
<td>QM001719307</td>
</tr>
</tbody>
</table>

**Table: BMC Atrium Discovery 10.2**

**Page 202 of 1671**
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Defect Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>15415</td>
<td><strong>Problem:</strong> Authentication cookie was not flagged as HTTPOnly.</td>
<td>ISS03662285, QM001676012</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Fixed in the code.</td>
<td></td>
</tr>
<tr>
<td>15434</td>
<td><strong>Problem:</strong> Issue while upgrading to version 8.3 where fiber channel HBA nodes were missing.</td>
<td>QM001728184</td>
</tr>
<tr>
<td></td>
<td><strong>Resolution:</strong> Fixed in the code.</td>
<td></td>
</tr>
</tbody>
</table>

**Defects Resolved in BMC Atrium Discovery Version 8.3**

<table>
<thead>
<tr>
<th>Defect Number</th>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>QM001667697</td>
<td>3781</td>
<td><strong>Problem:</strong> Use of &quot;which&quot; in discovery script breaks on some servers. <strong>Resolution:</strong> Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>10232</td>
<td></td>
<td><strong>Problem:</strong> Appliance Baseline page needs sort order. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>11855</td>
</tr>
<tr>
<td>SF13103</td>
<td>10408</td>
<td><strong>Problem:</strong> When an IP range list is &quot;hidden&quot; you cannot click to see the full list. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>SF9555</td>
<td>10537</td>
<td><strong>Problem:</strong> Useful virtualisation reports needed. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Uptime not extracted from SystemInfo or Srvinfo calls. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>22976</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> SNMP is failing on Netware 5.70.05. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>16178</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Core dump during shutdown - Discovery. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>21247, 22334, 22339</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Broadcom NIC not detected on Windows. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>22198</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Privileged execution of getDirectoryListing, getFileMetaData and getFileContent. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>22238</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Need a &quot;Lack of credential&quot; error message. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>22267</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Offline compaction fails due to a pattern incorrectly storing a node reference on a Host node. <strong>Workaround:</strong> Do not perform the compaction with the recode option.</td>
<td>22679</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> Export to Oracle fails when columns have spaces. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>22897</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Problem:</strong> You cannot filter using the third column of the Installed Packages report. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001673044</td>
<td>12952</td>
<td><strong>Problem:</strong> Slave purge button inconsistency. <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03565196</td>
</tr>
<tr>
<td>13115</td>
<td></td>
<td></td>
<td>ISS03596673</td>
</tr>
<tr>
<td>Defect Number</td>
<td>ID</td>
<td>Details</td>
<td>Customer Case</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
|              | TP13281| **Problem:** The date selector in the Report Builder does not honor the user's date preference setting.  
**Resolution:** Code fix. |                |
| QM001665930  | 13360  | **Problem:** Swap allocation on appliances should be increased to 8GB.  
**Resolution:** Code fix, this no longer occurs. | ISS03631844    |
|              | 13402  | **Problem:** RPMs should be installed for kdump use.  
**Resolution:** Code fix, this no longer occurs. | ISS03710031    |
|              | 13472  | **Problem:** After certain Discovery scans are started, runs are stopping at 99% completion.  
**Resolution:** Code fix, this no longer occurs. | ISS03637997, ISS03638945, ISS03622040, ISS03652843, ISS03632994. |
|              | 13482  | **Problem:** Stopping discovery while the discovery of a network device is in progress can take a long time. The device might be rediscovered when discovery is restarted.  
**Resolution:** Code fix, this no longer occurs. |                |
|              | 13493  | **Problem:** Discovering a large network device can take longer than the default reasoning timeout (30 minutes).  
**Resolution:** Code fix, this no longer occurs. |                |
|              | 13660  | **Problem:** The upgraded Admin group does not have permission to view the Export or CMDB-Sync pages.  
**Resolution:** Code fix, this no longer occurs. |                |
| QM001680848  | 13682  | **Problem:** Mainframe scan ending with reason "Unknown error".  
**Resolution:** Code fix, this no longer occurs. | ISS03680154    |
| QM001693484  | 13696  | **Problem:** Visualisation not updated for MainFrame.  
**Resolution:** Code fix, this no longer occurs. |                |
| QM001700324  | 13700  | **Problem:** Technical difficulties page on completion of gather.  
**Resolution:** Code fix, this no longer occurs. | ISS03712257, ISS03750161 |
| QM001674953  | 13731  | **Problem:** Transaction logs from datastore compaction can fill up the whole partition.  
**Resolution:** Code and documentation fix. | ISS03658956, ISS03681960 |
| QM001677253  | 13750  | **Problem:** An excessively long "where" clause in a query generates an unhelpful error CORBA.COMPLETED_MAYBE.  
**Resolution:** Code fix, this no longer occurs. | ISS03666994    |
| QM001667859  | 13764  | **Problem:** tw_injectip --replace does not work as documented.  
**Resolution:** Documentation fix. | ISS03636264    |
|              | 13768  | **Problem:** Solaris prtdiag parse failure results in number of physical and logical processors being reported incorrectly.  
**Resolution:** Code fix, this no longer occurs. | ISS03652666    |
<p>| QM001679007  | 13818  | <strong>Problem:</strong> | ISS03673992  |</p>
<table>
<thead>
<tr>
<th>Defect Number</th>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>13853</td>
<td></td>
<td><strong>Problem:</strong> Appliance baseline gives spurious warning about ntpd and vmware-tools at runlevel 5.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03676415</td>
</tr>
<tr>
<td>QM001680178</td>
<td>13899</td>
<td><strong>Problem:</strong> Usage notes (--help) misleading for the tw_cmdb_export2sync tool.</td>
<td>ISS03678004</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>13960</td>
<td></td>
<td><strong>Problem:</strong> For OpenVMS discovery, show network parsing fails if hostname is not known.</td>
<td>ISS03677985</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>13971</td>
<td></td>
<td><strong>Problem:</strong> Security vulnerability in returnURL redirect parameter.</td>
<td>ISS03662280</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>13982</td>
<td></td>
<td><strong>Problem:</strong> HBAs not detected on AIX, because of discovery parsing issue.</td>
<td>ISS03672449</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>TP13982</td>
<td></td>
<td><strong>Problem:</strong> When you run a report that has relationships in the results, such as the Installed Packages report, the New Packages discovered on Hosts report, and the New Patches discovered on Hosts report, the Query Builder is not displayed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001670520</td>
<td>14004</td>
<td><strong>Problem:</strong> Discovery Access document should explain the meaning of the runCommand status.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>14028</td>
<td></td>
<td><strong>Problem:</strong> Cisco Nexus Network switches show wrong serial number.</td>
<td>ISS03680887</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001678691</td>
<td>14029</td>
<td><strong>Problem:</strong> Bonded interfaces report nonsensical speed/duplex settings.</td>
<td>ISS03694229</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001685207</td>
<td>14035</td>
<td><strong>Problem:</strong> The gather page in the user interface does not collect session logs.</td>
<td>ISS03683463</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001685002</td>
<td>14048</td>
<td><strong>Problem:</strong> Reconciliation job is causing duplicate records as Domain field is not populated.</td>
<td>ISS03678131</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001685621</td>
<td>14064</td>
<td><strong>Problem:</strong> When a program running via RemCom crashes or initiates a UI pop-up on target machine, RemCom is blocked.</td>
<td>ISS03674449</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001681080</td>
<td>14073</td>
<td><strong>Problem:</strong> When a UNIX device with an unsupported shell is scanned, it fails with a timeout.</td>
<td>ISS03669062</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001682098</td>
<td>14074</td>
<td><strong>Problem:</strong> When discovering an HP-UX host, parsing getInterface list does not always find the speed, the duplex and negotiation mode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>14098</td>
<td></td>
<td><strong>Problem:</strong> Problem with MDZ service point caching.</td>
<td>ISS03696044</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001686473</td>
<td>14109</td>
<td><strong>Problem:</strong></td>
<td></td>
</tr>
</tbody>
</table>

BMC Atrium Discovery 10.2  Page 205 of 1671
<table>
<thead>
<tr>
<th>Defect Number</th>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
</table>
|              | 14110 | **Problem:** CiscoWorks import fails when a speed is not specified for an entry.  
**Resolution:** Code fix, this no longer occurs. |               |
|              | 14120 | **Problem:** When a list view would only contain one entry, the UI displays the node view after showing the Ajax "spinner". When this occurs, you are unable to use the browser's back button to return to the previous page.  
**Resolution:** Code fix, this no longer occurs. |               |
|              | 14161 | **Problem:** CiscoWorks import fails to import the CSV file content when entries without IP addresses are detected.  
**Resolution:** Code fix, this no longer occurs. | ISS03676549   |
|              | 14190 | **Problem:** When running offline compaction, you cannot use the original datastore after preparation steps.  
**Resolution:** Code fix, this no longer occurs. | ISS03700422   |
|              | 14207 | **Problem:** The template_sql_asset_integration.tpl file cannot be activated in 8.2 because default connections are not supported.  
**Resolution:** Code fix. | ISS03705373   |
|              | 14220 | **Problem:** In the device filter, queries using LifecycleStatus and Location do not work.  
**Resolution:** Code fix. | ISS03699116   |
|              | 14222 | **Problem:** Discovery fails on OpenVMS because we add the NUL byte before any prompt string.  
**Resolution:** Code fix, this no longer occurs. |               |
|              | 14224 | **Problem:** Network device discovery error "Unable to get the deviceinfo: UNKNOWN_PythonException".  
**Resolution:** Code fix, this no longer occurs. | ISS03702019   |
|              | 14243 | **Problem:** Mainframe IMS Database information discovery incomplete.  
**Resolution:** Code fix, this no longer occurs. | ISS03693787   |
|              | 14251 | **Problem:** Stop All Scans button does not work when defunct discovery processes exist.  
**Resolution:** Code fix, this no longer occurs. |               |
|              | 14259 | **Problem:** WMI arguments can be truncated.  
**Resolution:** Documentation fix. See WMI arguments |               |
|              | 14268 | **Problem:** Poor discovery performance and appliance errors from SNMP discovery.  
**Resolution:** Code fix, this no longer occurs. | ISS03704879   |
|              | 14285 | **Problem:** Upgrade script should check that there is available space in /var before upgrade.  
**Resolution:** Code fix, this no longer occurs. |               |
|              | 14300 | **Problem:** Create Integration point link should be removed.  
**Resolution:** Code fix, this no longer occurs. |               |
<p>|              | 14333 | <strong>Problem:</strong> | ISS03722398   |</p>
<table>
<thead>
<tr>
<th>Defect Number</th>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>QM001680931</td>
<td>14358</td>
<td><strong>Problem:</strong> BMC Atrium Discovery to BMC Atrium mapping requires additional dependency between Mainframe and BAI.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001692734</td>
<td>14359</td>
<td><strong>Problem:</strong> Cannot discover the CPU/RAM information on HP-UX.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03678094</td>
</tr>
<tr>
<td>QM001686848</td>
<td>14381</td>
<td><strong>Problem:</strong> Mainframe extension for CMDB incorrectly sets audittype=none for BMC_BaseElement Core CDM attributes.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03700534</td>
</tr>
<tr>
<td>QM001696908</td>
<td>14393</td>
<td><strong>Problem:</strong> Editing a Location node in the UI is not working as expected.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001686895</td>
<td>14401</td>
<td><strong>Problem:</strong> Discovery is failing in getInterfaceList on AIX 4.2.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001697505</td>
<td>14406</td>
<td><strong>Problem:</strong> In Mainframe JVM Settings increase maximum memory limit to 1024MB.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03712888</td>
</tr>
<tr>
<td>QM001697506</td>
<td>14408</td>
<td><strong>Problem:</strong> For &quot;Java heap space&quot; error is misleading, should read &quot;Out of memory error - Java heap space&quot;.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03712888</td>
</tr>
<tr>
<td>QM001696447</td>
<td>14427</td>
<td><strong>Problem:</strong> The tideway.py file is showing as deprecated in the appliance startup.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03736960</td>
</tr>
<tr>
<td>QM001700149</td>
<td>14448</td>
<td><strong>Problem:</strong> Failure in generic search caused by processwith.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001699198</td>
<td>14468</td>
<td><strong>Problem:</strong> Xen host with &quot;aix&quot; in the hostname causes AIX discovery scripts to be used for discovery.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03766011 ISS03745096</td>
</tr>
<tr>
<td>QM001697510</td>
<td>14473</td>
<td><strong>Problem:</strong> Uninformative error messages in UI when CMDB mapping patterns are not activated.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001697909</td>
<td>14474</td>
<td><strong>Problem:</strong> The remcomsv.exe file should be removed from Windows targets.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001701089</td>
<td>14477</td>
<td><strong>Problem:</strong> The Observed Communications report fails.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001693571</td>
<td>14500</td>
<td><strong>Problem:</strong> Zebra printer with sysoid : 1.3.6.1.4.1.683.6 incorrectly discovered.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03725047</td>
</tr>
<tr>
<td>QM001687441</td>
<td>14520</td>
<td><strong>Problem:</strong> OpenVMS process names including spaces cannot be parsed correctly.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03701751</td>
</tr>
<tr>
<td>QM001702664</td>
<td>14536</td>
<td><strong>Problem:</strong> Discovery fails on Solaris 10 when sudo requires password for PRIV_PS.  &lt;br&gt; <strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03749573</td>
</tr>
<tr>
<td>Defect Number</td>
<td>ID</td>
<td>Details</td>
<td>Customer Case</td>
</tr>
<tr>
<td>---------------</td>
<td>-------</td>
<td>------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>QM001671087</td>
<td>14554</td>
<td>Problem: The /var directory fills with cron emails preventing Services from restarting. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001670514</td>
<td>14563</td>
<td>Problem: Non-existent commands are reported as &quot;OK&quot; in the DiscoveryAccess RunCommand status. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03646411</td>
</tr>
<tr>
<td>QM001703929</td>
<td>14578</td>
<td>Problem: SYSPLEX information not shared in environment where a sysplex spans multiple mainframes. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03756625</td>
</tr>
<tr>
<td>QM001702839</td>
<td>14595</td>
<td>Problem: Documentation should explain the concept of root nodes and why new custom nodes are not automatically exported. <em>Resolution:</em> Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001704455</td>
<td>14602</td>
<td>Problem: Discovery of OpenVMS version 8.4 fails at getInterfaceList script due to unequal lengths of Autonegotiation entries. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03750864</td>
</tr>
<tr>
<td>QM001705138</td>
<td>14603</td>
<td>Problem: Asynchronous search can be incorrectly cancelled during blocking getResults calls. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001703740</td>
<td>14607</td>
<td>Problem: From HP-UX version 11.23, &quot;parstatus&quot; in the host_info script cannot find the CPU information. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001689493</td>
<td>14614</td>
<td>Problem: Need script to remove &quot;dark space&quot; - tw_remove_darkspace. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03704879</td>
</tr>
<tr>
<td>QM001704659</td>
<td>14623</td>
<td>Problem: Categorization for BMC_IPENDPOINT and BMC_LANENDPOINT submitted to CMDB is incorrect. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001704880</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QM001709102</td>
<td>14642</td>
<td>Problem: Need to add settings to DB_CONFIG file to prevent db_recover running out of memory. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03772783</td>
</tr>
<tr>
<td></td>
<td>14644</td>
<td>Problem: Windows hosts with identical serial number are very likely to be merged. Resolution: Code fix, this no longer occurs.</td>
<td>ISS03735278</td>
</tr>
<tr>
<td>QM001685174</td>
<td>14662</td>
<td>Problem: Poorly positioned and immovable dialog boxes on restore from backup. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001706866</td>
<td>14668</td>
<td>Problem: Need to alter the perception of the system to an appliance rather than a general purpose OS. Resolution: Code and documentation fix.</td>
<td></td>
</tr>
<tr>
<td>Defect Number</td>
<td>ID</td>
<td>Details</td>
<td>Customer Case</td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>------------------------------------------------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>QM001706440</td>
<td>14672</td>
<td><strong>Problem:</strong> Relationship between CICS and Sysplex not being shown.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001707516</td>
<td>14686</td>
<td><strong>Problem:</strong> CMDB sync mapping incorrect for Host with processors with</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>multiple CPU types and speeds.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001707001</td>
<td>14713</td>
<td><strong>Problem:</strong> Processor sync mapping to CMDB (CMDB.Host_processor) does</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>not support PowerPC 6 and 7.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001706278</td>
<td>14717</td>
<td><strong>Problem:</strong> Engine and pattern performance information not available at</td>
<td></td>
</tr>
<tr>
<td>QM001707286</td>
<td></td>
<td>all Reasoning log levels.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001706357</td>
<td>14723</td>
<td><strong>Problem:</strong> The <code>tw_purge_slave_logs</code> tool fails when run with the</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>--oldest-days=NUM</code> option.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001706655</td>
<td>14731</td>
<td><strong>Problem:</strong> DB Cache size setting in UI can be set too high.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001698828</td>
<td>14754</td>
<td><strong>Problem:</strong> Patch name reported on Solaris systems where no patches are</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>installed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001709071</td>
<td>14756</td>
<td><strong>Problem:</strong> For HP-UX platforms, <code>getInterfaceList</code> uses <code>netstat -in</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>whereas <code>netstat -inw</code> is sometimes required.</td>
<td>ISS03790271</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001693910</td>
<td>14796</td>
<td><strong>Problem:</strong> The <code>weblogic JDBC resource</code> is not found when the URL leads</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>to an aliased database.</td>
<td>ISS03692485</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001720262</td>
<td>14828</td>
<td><strong>Problem:</strong> <code>CapabilityList</code> for <code>BMC_ComputerSystem</code> and <code>BMC_Mainframe</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>is set to enum value rather than name.</td>
<td>ISS03835157,</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td>ISS03836568</td>
</tr>
<tr>
<td>QM001708299</td>
<td>14836</td>
<td><strong>Problem:</strong> Discovery of HP-UX 11.31 Itanium systems does not populate</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RAM or Processor details.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001711796</td>
<td>14841</td>
<td><strong>Problem:</strong> The <code>netadmin</code> user should be the recommended way of configuring</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>network settings.</td>
<td>ISS03806752</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001711188</td>
<td>14844</td>
<td><strong>Problem:</strong> Upgrade resets the time zone settings to BST.</td>
<td>ISS03781188</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code and documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001680947</td>
<td>14917</td>
<td><strong>Problem:</strong> Export log file name, <code>tw_cmdb-export.log</code> is confusing,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>should be renamed <code>tw_export.log</code>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001707385</td>
<td>14928</td>
<td><strong>Problem:</strong> Discovery stays logged in on target regardless of scan timeout</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>value.</td>
<td>ISS03778681</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Resolution:</strong> Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001705323</td>
<td>14929</td>
<td><strong>Problem:</strong></td>
<td>ISS03767301</td>
</tr>
<tr>
<td>Defect Number</td>
<td>ID</td>
<td>Details</td>
<td>Customer Case</td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>QM001664211</td>
<td>14941</td>
<td>Problem: Synchronizing into a new custom CMDB attribute is not possible unless the tideway service is restarted. Resolution: Code and documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001711782</td>
<td>14946</td>
<td>Problem: CMDB Sync mappings: Name needs valid value for BMC_IPEndpoint. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001690038</td>
<td>14949</td>
<td>Problem: Mainframe discovery - Insufficient user access reported only in logs. Should be reported as script failures on Discovery Access page. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001716540</td>
<td>14962</td>
<td>Problem: Null Pointer Exceptions when using CMDB/API 7.6, but not when using CMDB/API 2.0 during CMDB Sync. Resolution: Code and documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001716869</td>
<td>14982</td>
<td>Problem: Traceback when applying sensitive data filters. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001700541</td>
<td>15006</td>
<td>Problem: How many threads should be specified when setting up the CMDB sync connection? Resolution: Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001690098</td>
<td>15007</td>
<td>Problem: Documentation does not provide enough details about the permission of the database credentials. Resolution: Documentation fix.</td>
<td></td>
</tr>
<tr>
<td>QM001719571</td>
<td>15030</td>
<td>Problem: AIX Host package_count differs from the count of packages. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001703981</td>
<td>15042</td>
<td>Problem: Technical difficulties page on LDAP Configuration when an invalid CA Certificate is uploaded. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001719109</td>
<td>15054</td>
<td>Problem: Excessive memory consumption when a very large list of IP addresses is entered for scanning. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001712384</td>
<td>15061</td>
<td>Problem: DDD removal option &quot;never&quot; should only be used under guidance from Customer Support. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001714773</td>
<td>15062</td>
<td>Problem: The tw_supportability script should have timestamps. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001718966</td>
<td>15064</td>
<td>Problem: An ssh timeout caused by &quot;PASSWORD&quot; instead of &quot;Password&quot;. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001719228</td>
<td>15083</td>
<td>Problem: Typo in upgrade script. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001719109</td>
<td>15100</td>
<td>Problem: Data store unnecessarily re-indexes unchanged attributes. Resolution: Code fix, this no longer occurs.</td>
<td></td>
</tr>
<tr>
<td>QM001718878</td>
<td>15118</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Known defects in this version

The table below describes the defects in this version, 10.2.0.1, of BMC Atrium Discovery.

<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-16407</td>
<td><strong>Problem:</strong> Missing item in XML data while scanning Cisco CIMC device.</td>
<td>QM001883474</td>
</tr>
<tr>
<td>ADDM-16250</td>
<td><strong>Problem:</strong> Sometimes, CMDB resync logs messages for deleted connections.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16246</td>
<td><strong>Problem:</strong> The following error, “ERROR: Failed to remove sync queue persist directory” may be logged when upgrading.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16239</td>
<td><strong>Problem:</strong> Occasionally, when viewing a CMDB sync candidate set, the following traceback is displayed in the UI: “SyncTargetTabbedPage.ajaxUpdates: 'NoneType' object has no attribute 'get'”</td>
<td></td>
</tr>
<tr>
<td>ADDM-16236</td>
<td><strong>Problem:</strong> CMDB resync page sometimes displays the Technical Difficulties page when other operations (pattern deletion) take place.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16235</td>
<td><strong>Problem:</strong> CMDB Sync Filtering page unnecessarily shows a &quot;Return to Previous Page&quot; link after filtering on a condition with no results.</td>
<td></td>
</tr>
<tr>
<td>ADDM-16233</td>
<td><strong>Problem:</strong> Amending a CMDB Sync connection to an empty data set does not report ‘ready’ correctly.</td>
<td></td>
</tr>
</tbody>
</table>
| ADDM-16232 | **Problem:** Upgrade on cluster may fail to start properly: "Cluster startup checks failed: Incompatible product versions."   
Workaround: Start the cluster service and then the tideway service on any members that have logged the "Incompatible product versions" message. |               |
| ADDM-16227 | **Problem:** Occasionally nodes queued during a CMDB resync remain after the resync ends.                                                   
Workaround: Manually sync another node to clear the queue. |               |
| ADDM-16223 | **Problem:** IE 10 and IE11: CMDB sync filtering defined filters do not show up.                                                            
Workaround: Use an alternative browser.    |               |
| ADDM-16222 | **Problem:** IE 8: CMDB sync filter selecting options has a delay.                                                                           
Workaround: Use an alternative browser.    |               |
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
</table>
| ADDM-16209   | **Problem:** Deleting a sync connection during a cluster rebalance causes “WARNING: Failed to collect node counts for rebalance estimates - will retry shortly.” messages to be logged.  
**Workaround:** Avoid deleting sync connections during cluster operations that cause a rebalance. |              |
| ADDM-16208   | **Problem:** Replacing a CMDB sync connection with an identical one but with the data model changed to 5 (a deprecated model), results in the CMDB arserver service to fail. Symptoms are that the resync-commit status display stops updating after showing that relationships are being created.  
**Workaround:** Restart the CMDB arserver service.  
  * To sync again with the original data model, delete the CMDB sync connection, restart the tideway services and add a new connection.  
  * To sync with the changed data model (data model 5), delete the CMDB sync connection, clean the dataset (see below), restart the tideway services and add a new connection.  
  * To clean the dataset, use the corresponding option in the cmdbdiag.exe utility, or create a Reconciliation job to “delete” the dataset. |              |
| ADDM-16203   | **Problem:** The tw_sync_control --resync-commit does not find the resync ID.  
**Workaround:** Perform the resync using the UI. |              |
| ADDM-16195   | **Problem:** Some REMOTE and EXPORTED FileSystem nodes are not linked.                                                                                                                                 | QM001878136  |
| ADDM-16193   | **Problem:** Resync failing to unset attribute on CI where attribute has been deleted on node in datastore.                                                                                          |              |
| ADDM-15866   | **Problem:** Viewing an SI pattern generated from a Discovered Service shows all services, not just matching ones.  
**Workaround:** None.                                                                 |              |
| ADDM-15703   | **Problem:** Datastore rebalance hangs on a node with state over 2GB.  
**Workaround:** None.                                                                                                                   | QM001860421  |
| ADDM-15341   | An appliance may suggest applying an already applied upgrade.  
**Workaround:** None.                                                                                                                    |              |
| ADDM-14921   | No upgrade status displayed after reboot.  
**Workaround:** None.                                                                                                                      |              |
| ADDM-14486   | Credential testing should check open ports.  
**Workaround:** None.                                                                                                                      |              |
| ADDM-14360   | **Problem:** When you re-provision a NIC (for example, change the interface type on a virtual machine) and reboot the system for the changes to take effect, the NIC is provisioned as eth1 rather than eth0, which BMC Atrium Discovery requires.  
**Workaround:** See Appliance does not have an IP on eth0. |              |
| ADDM-14209   | **Problem:** You cannot log in to an https enabled appliance using the Safari browser. A security error dialog is displayed.  
**Workaround:** None.                                                                                                                     |              |
| ADDM-13825   | **Problem:** If graph definitions refer to a node kind that is not in the taxonomy, the visualizations fail to start. The logging shows that the problem is related to the taxonomy, though this can be difficult to find as there is much unrelated information.  
**Workaround:** Remove the node kind from the graph definition. |              |
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
</table>
| ADDM-13720  | **Problem:** In some situations a Windows host might take many hours to discover. Where this occurs, view the corresponding DiscoveryAccess and select the Discovery Method Timings report from the Reports drop-down. You might see a series of Discovery Method Duration entries lasting 29 to 30 minutes and then some shorter ones of up to a few seconds in duration. Where this occurs, the worker process might have been discarded and a new one created. Consequently, a large number of defunct worker processes are visible in the Windows Task Manager. For each of these processes (when at DEBUG log level) the last entry in the log file is of the form:  


You will also see entries in the manager process log file of the form:

"Exception checking worker alive:
CORBA.TRANSIENT(omniORB.TRANSIENT_CallTimedout,
CORBA.COMPLETED_MAYBE) 1784: 2012-01-30
21:49:26,351: discovery.slave.manager.workers:
INFO: Dropping worker IOR:010000002f"

This defect is also present in BMC Atrium Discovery 8.3 and 8.3 SP1. **Workaround:** Rescan the host. If the number of defunct processes becomes excessive, you must reboot the host to remove them. | None.                                |
<p>| ADDM-13579  | <strong>Problem:</strong> If you try and extract (running the upgrade script with the --upgrade option) the contents of the upgrade package on a host that does not have BMC Atrium Discovery installed, the script returns errors. <strong>Workaround:</strong> None. |               |
| ADDM-13568  | <strong>Problem:</strong> When a CMDB sync connection is configured with a user with minimal permissions, you might not be able to connect if the user has open connections from other locations. <strong>Workaround:</strong> Use a user with at least the CMDB Demo user permissions. |               |
| ADDM-13543  | <strong>Problem:</strong> The name server retries option that can be set the Administration &gt; Appliance Configuration &gt; Name Resolution page has no effect. It will always attempt 3 retries. <strong>Workaround:</strong> None. |               |
| ADDM-13435  | <strong>Problem:</strong> In the Collaborative Application Mapping (CAM) user interface, using a named value that has a list in the results results in a traceback. <strong>Workaround:</strong> None. |               |
| ADDM-13415  | <strong>Problem:</strong> The wiki markup in the CAM text editor does not render correctly in the PDFs. <strong>Workaround:</strong> None. | QM001720259   |
| ADDM-13398  | <strong>Problem:</strong> AD Proxy shows the confusing log reference of &lt;no username&gt; in logs. <strong>Workaround:</strong> None. | QM001718958   |
| ADDM-13366  | <strong>Problem:</strong> Attempting to discover a host server running version 2.5 of the optional MainView for WebSphere Application Server product results in the script failure &quot;message View 492 not available&quot;, and no WebSphere Discovered Application Components are created, preventing extended discovery. <strong>Workaround:</strong> None. |               |
| ADDM-13358  | <strong>Problem:</strong> Instrument the scanning appliance to capture unexpected exceptions while sending data to the consolidator. <strong>Workaround:</strong> None. | QM001718958   |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Details</th>
<th>Customer Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDM-13318</td>
<td>Problem: When the IP list for a host changed dramatically, on the consolidator no Endpoint Identity was created.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-13294</td>
<td>Problem: The user interface displays a &quot;The appliance is having technical difficulties with this page&quot; error if you add a login credential that includes a control character ASCII 0-31.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-13237</td>
<td>Problem: Processor sync mapping to CMDB (CMDB.Host_processor) does not support PowerPC 6 and 7 for BMC_Processor.ProcessorFamily.</td>
<td>QM001718170</td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-13011</td>
<td>Problem: Cannot move proxy pools with spaces in their names using actions menu.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-12895</td>
<td>Problem: &quot;ksh[n]: tw_report: not found&quot; appended to the output of many Discovered results for Solaris 10.</td>
<td>ISS03729494</td>
</tr>
<tr>
<td></td>
<td>Workaround: Set force subshell = True for that login credential.</td>
<td></td>
</tr>
<tr>
<td>ADDM-12421</td>
<td>Problem: TCPvcon cannot be pushed to Windows 2000 hosts.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: Deploy the utility manually.</td>
<td></td>
</tr>
<tr>
<td>ADDM-11391</td>
<td>Problem: Wrong RAM reported by WMI against Windows 2008 Server VMs on Hyper-V.</td>
<td>QM001732452</td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-11341</td>
<td>Problem: Errors appear in ECA logs &quot;Errors not related to a node&quot;.</td>
<td>QM001703605</td>
</tr>
<tr>
<td>ADDM-11339</td>
<td>Problem: Log rolling failure in Windows Proxy results in traceback.</td>
<td>21992</td>
</tr>
<tr>
<td>ADDM-11239</td>
<td>Problem: If the &quot;Only show working set in visualizations&quot; menu option is used while viewing a visualization, the visualization does not always redraw correctly to honor the new option state.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: Manually refresh the browser page.</td>
<td></td>
</tr>
<tr>
<td>ADDM-11123</td>
<td>Problem: TRANSIENT_CallTimedout should be modified slightly for user display.</td>
<td>QM00021758</td>
</tr>
<tr>
<td></td>
<td>Workaround: None.</td>
<td></td>
</tr>
<tr>
<td>ADDM-11023</td>
<td>Problem: Discovery runs older than the ten most recent are not displayed well.</td>
<td>ISS03625937</td>
</tr>
<tr>
<td>ADDM-9922</td>
<td>Problem: TPL activation doesn't handle special characters well, for example a back tick.</td>
<td>QM001749274</td>
</tr>
<tr>
<td>ADDM-9734</td>
<td>Problem: When connecting to BMC Atrium Discovery via HTTPS using an alias hostname, you are prompted for verification.</td>
<td>QM00013175</td>
</tr>
<tr>
<td>ADDM-9497</td>
<td>Problem: History comparison is unusable with more than a small number of entries.</td>
<td>8444</td>
</tr>
<tr>
<td>ADDM-8468</td>
<td>Problem: When doing an snmpGet in a pattern, it is possible that some SNMP agents for certain OIDs will return data for a different OID than the one requested. Because the OID for the data retrieved doesn't match the OID that the request was for, the corresponding attribute will not be set.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Workaround: The mapping can be set to include the requested OID and the OID that will actually be returned. For example:</td>
<td></td>
</tr>
</tbody>
</table>
|          | table oid_map 1.0
|          | ".1.3.6.1.4.1.11.2.7.1.75.8.10.10.10.1" -> "wont_get_this";                                                                                                                                        |               |
## Important information for users of BMC Atrium CMDB

To integrate BMC Atrium Discovery with BMC Atrium CMDB, there are some prerequisites, known issues, and limitations that you must be aware of.

### Interaction of BMC Atrium Discovery with other BMC data providers

BMC Atrium Discovery can co-exist with other BMC data providers. To prevent any potential duplicated Configuration items (CIs) in the BMC Atrium CMDB Asset dataset, BMC recommends the following minimum product versions when using other data providers with BMC Atrium Discovery:

- **BMC Atrium CMDB**: At the time of its release, integration of the current version of BMC Atrium Discovery was supported with the following versions of BMC Atrium CMDB:
9.0 – Fully supported. The integration is fully supported, though formal BMC certification is pending. There are some performance issues between BMC Atrium Discovery 10.2 and CMDB 9.0. These are resolved in CMDB 9.0 SP1.

- 8.1.02
- 8.1.01
- 8.1.00
- 8.0.00
- 7.6.04

If you intend to use CMDB sync, BMC recommends that you use BMC Atrium Discovery version 10.2.0.2 or later.

For more information, check the BMC Solution and Product Availability and Compatibility utility (SPAC) (support login required).

- **BMC BladeLogic Server Automation**: BMC BladeLogic Server Automation 8.0 SP2. For more details, see the BladeLogic section.
- **BMC Performance Manager (BPM) and BMC Performance Assurance (BPA)**: BMC Performance Manager Portal 2.7.00.040 is recommended. For more details, see the BPM /BPA section.
- **BMC BladeLogic Client Automation (BBCA)**: BMC BladeLogic Client Automation 8.1.01 is recommended. Version 8.1.01 contains the fix of two known issues. For more details, see the BBCA section.

**Changed behavior in BMC Atrium Discovery 10.1 and later**

In previous releases and with TKU 2014-Nov-1 onwards installed, when BMC Atrium Discovery encountered NetApp devices, they were modeled using NetworkDevice nodes. The default CDM mappings allowed these to be synchronized to the CMDB. With the introduction of storage discovery, they are now modeled as StorageDevice nodes. The default CDM mappings only synchronize storage data when StorageSystem nodes exist, which are only produced by the Storage TKU.

As a consequence, in BMC Atrium Discovery 10.1, NetApp devices are only synchronized to CMDB if the Storage TKU is installed.

**Prerequisites for CMDB synchronization**

Before you can synchronize data to a supported BMC Atrium CMDB version, you must complete the following required tasks in the given order:

**Integrating with earlier versions of BMC Atrium CMDB**

BMC Atrium CMDB versions earlier than 7.6.04 are no longer supported and are not recommended for use. Synchronizing data to these earlier versions required you to complete some additional tasks. For more information, see the BMC Atrium Discovery 8.3 documentation.
1. **Create the BMC.ADDM dataset**: Manually create the BMC.ADDM dataset in the CMDB before attempting synchronization.

2. **Create the Job to merge the BMC.ADDM dataset with BMC.ASSET**: Create the job to reconcile the BMC.ADDM dataset with the BMC.ASSET dataset.

3. **Check the BMC.ADDM dataset configuration**: When using ITSM 7.0 with a backward compatibility patch, you must ensure that the BMC.ADDM dataset is trusted.

After completing the preceding tasks, you can start synchronizing BMC Atrium Discovery data to BMC Atrium CMDB.

**Performance considerations**

To obtain the maximum synchronization performance when using CMDB synchronization with BMC Atrium Discovery version 8.3.00 and later, you should consider tuning the database which BMC Atrium CMDB (or BMC Remedy AR System) is using. For more information, see the following documentation corresponding to your product version:

- BMC Remedy AR System 8.1 Performance tuning for Business Service Management
- BMC Remedy AR System 8.0 Performance tuning for Business Service Management
- BMC Remedy AR System Server 7.6 Performance Tuning for Business Service Management White Paper

The following sections in the White Paper are particularly relevant:

- Tuning an Oracle server.
- Best practices for tuning Oracle database servers.
- Tuning a SQL Server database.
- Best practices for tuning SQL Server database servers.

**Co-existence with other data providers**

The following issues exist in the corresponding BMC product that can lead to duplicate CIs in the BMC Atrium CMDB ASSET dataset when reconciling with BMC Atrium Discovery:

⚠️ **Fixes which are available as hot fixes will be rolled into future patches. BMC recommends you to see the latest patch release notes for the defect number to see if the hot fix is already included.**

**BladeLogic**

The minimum product version recommended is BladeLogic Server Automation 8.0 SP2 with fixes for the following issues and defects.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW00355356</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Issue | Description | Fix
---|---|---
 | Serial number is a strong indicator of identity for the BMC_ComputerSystem and is therefore used in reconciliation. On older Solaris Sparc platforms serial number is not available. BL would set the serial number to a value derived from hostid of the BMC_ComputerSystem class. In this situation duplicate BMC_ComputerSystem CIs would be created. | Contact BMC Customer Support requesting BladeLogic Server Automation 8.0 SP2
SW00355142 | For Windows Hosts BladeLogic sets the Domain attribute for BMC_ComputerSystem to be the Windows Domain and not its DNS Domain. The Domain attribute forms a part of the TokenID for a BMC_ComputerSystem CI. In this situation duplicate BMC_ComputerSystem CIs will be created. | Contact BMC Customer Support requesting Hot Fix for defect SW00355142
SW00355359 | isVirtual attribute incorrectly set to No for Physical system. In this situation duplicate BMC_ComputerSystem CIs will be created. | Contact BMC Customer Support requesting Hot Fix for defect SW00355359
QM001646701 | When executing the BL reconciliation job after the BMC Atrium Discovery reconciliation job the CIs associated with a merged BMC_ComputerSystem class (for example: BMC_Processor and BMC_IPEndpoint) are not merged and result in duplicate CIs. | Contact BMC Customer Support requesting Hot Fix for defect QM001646701
QM001642175 | Server name should be picked up from FQ_HOST instead of Host property on its product. When a ComputerSystem is discovered by both BladeLogic and BMC Atrium Discovery this issue can result in the ComputerSystem not being reconciled when the case of the Hostname is not the same for the data from the two products. | Contact BMC Customer Support requesting BladeLogic Server Automation 8.0 SP2

**BPM/BPA**

The minimum product version recommended is BMC Performance Manager Portal 2.7.00.040.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Fix</th>
</tr>
</thead>
</table>
QM001609002 | BPM has no concept of Virtual Hosts. Therefore, when it sets TokenID for a BMC_ComputerSystem it uses the format defined for Physical Hosts. The reconciliation of this data with data from other providers that are aware of virtualization will fail if you reconcile the BPM data after another discovery provider. This will result in duplicate CIs. | BMC Performance Manager Portal 2.7.00.040

**BBCA**

The minimum product version recommended is BMC BladeLogic Client Automation 8.1.01.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Fix</th>
</tr>
</thead>
</table>
SW00355355 | Serial number is a strong indicator of identity for the BMC_ComputerSystem and is therefore used in reconciliation. On older Solaris Sparc platforms serial number is not available. BBCA would set the serial number to a value derived from hostid of the BMC_ComputerSystem class. In this situation duplicate BMC_ComputerSystem CIs would be created. |
<table>
<thead>
<tr>
<th>Issue</th>
<th>Description</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>SW00355361</td>
<td>isVirtual attribute incorrectly set to No for Physical system. In this situation duplicate BMC_ComputerSystem CIs will be created.</td>
<td>Fixed in BMC BladeLogic Client Automation 8.1.01.</td>
</tr>
</tbody>
</table>
Accessing the release

Supported versions of BMC Atrium Discovery

August 2014 Technology Knowledge Update and Operating System Upgrade packages

BMC Software announces the availability of the following product updates which support the corresponding versions of BMC Atrium Discovery:

Technology Knowledge Update (TKU) and Extended Data Pack:

- Supported versions: 8.3, 8.3 SP1, 8.3 SP2, 8.3 SP3, 9.0, 9.0 SP1, 9.0 SP2, and 10.0.
- **Release highlights**: Includes 8 new pattern modules for coverage of discovered Microsoft, Oracle, Teradata, IBM, *and* BMC products, 18 enhancements that strengthen discovery coverage, a network device integration module that contains 16 new network device definitions, 12 new printer definition, and it resolves many important defects.
- **Content details and installation instructions**: For information about new and enhanced pattern modules and download instructions, see the August 2014 Technology Knowledge Update page. To learn about network device integration module and the corresponding network device definitions, see the Network Device, Printer and SNMP Managed Device Definitions page.

Operating System (OS) Upgrade package:

- **Supported versions**: The RHEL 6 OS upgrade supports only the newly installed version 9.0, 9.0 SP1, 9.0 SP2, and 10.0 virtual appliances (on RHEL 6). The RHEL 5 OS upgrade supports versions 8.3, 8.3 SP1, 8.3 SP2, and 8.3 SP3 and the upgraded version 9.0, 9.0 SP1, and 9.0 SP2 virtual appliances (on RHEL 5).
- **Release highlights**: The RHEL 6 OS upgrades contain the applicable Red Hat packages until August 14, 2014 and the RHEL 5 OS upgrades contain the applicable Red Hat packages until August 14, 2014. The date corresponding to each OS upgrade reflects the release date of the latest Red Hat packages which are included in this month's OS upgrades.
- **Latest OS upgrade files**: The latest RHEL 6 OS upgrade file (for the 64 bit architecture) is `ADDM_OS_Upgrade_64_6.14.08.14_385644_ga.sh.gz`. The latest RHEL 5 OS upgrade files are `ADDM_OS_Upgrade_64_5.14.08.14_385617_ga.sh.gz` (for the 64 bit architecture) and `ADDM_OS_Upgrade_32_5.14.08.14_385617_ga.sh.gz` (for the 32 bit architecture).
- **Content details and upgrade instructions**: To learn about the OS upgrade packages installed on your appliance after the latest OS upgrade, see the Latest RHEL 5 operating system upgrade and Latest RHEL 6 operating system upgrade pages. For upgrade instructions, see Upgrading the operating system. To learn about the earlier OS upgrade packages, from Previous operating system upgrades, select the link for the corresponding period.
The above-referenced product update packages are available for download from the BMC Electronic Product Distribution (EPD) site. Each package can be installed independently. See Accessing the release for more information.

September 2014 Technology Knowledge Update and Operating System Upgrade packages

BMC Software announces the availability of the following product updates which support the corresponding versions of BMC Atrium Discovery:

Technology Knowledge Update (TKU) and Extended Data Pack:

- Supported versions: 8.3, 8.3 SP1, 8.3 SP2, 8.3 SP3, 9.0, 9.0 SP1, 9.0 SP2, 10.0, and 10.0 Patch 2.
- **Release highlights**: Includes 11 new pattern modules for coverage of discovered Oracle, Teradata, Puppet Labs, Sailpoint, Apache, and Microsoft products, 11 enhancements that strengthen discovery coverage, a network device integration module that contains new network device definitions, new printer definition, and it resolves many important defects.
- **Content details and installation instructions**: For information about new and enhanced pattern modules and download instructions, see the September 2014 Technology Knowledge Update page. To learn about network device integration module and the corresponding network device definitions, see the Network Device, Printer and SNMP Managed Device Definitions page.

Operating System (OS) Upgrade package:

**Important vulnerability fix inside**

September 2014 OS upgrade includes a Red Hat fix for a serious vulnerability in the Bash shell (CVE-2014-6271), that was publicly disclosed recently. Red Hat has provided an updated version of bash which solves the problem. We are providing an OS Update now to apply to your affected ADDM Appliances that fixes CVE-2014-6271. BMC Atrium Discovery appliances running Red Hat Linux v5 of v6 are affected. BMC strongly recommends you to upgrade the effected appliances. See the post in BMC Communities for more information.

- **Supported versions**: The RHEL 6 OS upgrade supports only the newly installed version 9.0, 9.0 SP1, 9.0 SP2, and 10.0 virtual appliances (on RHEL 6). The RHEL 5 OS upgrade supports versions 8.3, 8.3 SP1, 8.3 SP2, and 8.3 SP3 and the upgraded version 9.0, 9.0 SP1, and 9.0 SP2 virtual appliances (on RHEL 5).
• **Release highlights**: The RHEL 6 OS upgrades contain the applicable Red Hat packages until September 25, 2014 and the RHEL 5 OS upgrades contain the applicable Red Hat packages until September 25, 2014. The date corresponding to each OS upgrade reflects the release date of the latest Red Hat packages which are included in this month’s OS upgrades.

• **Latest OS upgrade files**: The latest RHEL 6 OS upgrade file (for the 64 bit architecture) is `ADDM_OS_Upgrade_64_6.14.09.25_393903 Ga.sh.gz`. The latest RHEL 5 OS upgrade files are `ADDM_OS_Upgrade_64_5.14.09.25_393937 Ga.sh.gz` (for the 64 bit architecture) and `ADDM_OS_Upgrade_32_5.14.09.25_393937 Ga.sh.gz` (for the 32 bit architecture).

• **Content details and upgrade instructions**: To learn about the OS upgrade packages installed on your appliance after the latest OS upgrade, see the Latest RHEL 5 operating system upgrade and Latest RHEL 6 operating system upgrade pages. For upgrade instructions, see Upgrading the operating system. To learn about the earlier OS upgrade packages, from Previous operating system upgrades, select the link for the corresponding period.

The above-referenced product update packages are available for download from the BMC Electronic Product Distribution (EPD) site. Each package can be installed independently. See Accessing the release for more information.

**October 2014 Technology Knowledge Update and Operating System Upgrade packages**

BMC Software announces the availability of the following product updates which support the corresponding versions of BMC Atrium Discovery:

**Technology Knowledge Update (TKU) and Extended Data Pack**:

• Supported versions: 8.3, 8.3 SP1, 8.3 SP2, 8.3 SP3, 9.0, 9.0 SP1, 9.0 SP2, 9.0 SP3, 10.0, and 10.0 Patch 2.

• **Release highlights**: Includes 7 new pattern modules for coverage of discovered Microsoft, Oracle, MemSQL, Pivotal, Opscode, Docker and opensource products, 14 enhancements that strengthen discovery coverage, a network device integration module that contains 13 new network device definitions, and it resolves many important defects.

• **Content details and installation instructions**: For information about new and enhanced pattern modules and download instructions, see the October 2014 Technology Knowledge Update page. To learn about network device integration module and the corresponding network device definitions, see the Network Device, Printer and SNMP Managed Device Definitions page.

**Operating System (OS) Upgrade package**:
- **Supported versions**: The RHEL 6 OS upgrade supports only the newly installed version 9.0, 9.0 SP1, 9.0 SP2, and 10.0 virtual appliances (on RHEL 6). The RHEL 5 OS upgrade supports versions 8.3, 8.3 SP1, 8.3 SP2, and 8.3 SP3 and the upgraded version 9.0, 9.0 SP1, and 9.0 SP2 virtual appliances (on RHEL 5).

- **Release highlights**: The RHEL 6 OS upgrades contain the applicable Red Hat packages until October 17, 2014 and the RHEL 5 OS upgrades contain the applicable Red Hat packages until October 17, 2014. The date corresponding to each OS upgrade reflects the release date of the latest Red Hat packages which are included in this month's OS upgrades.

- **Latest OS upgrade files**: The latest RHEL 6 OS upgrade file (for the 64 bit architecture) is ADDM_OS_Upgrade_64_6.14.10.17_397388.ga.sh.gz. The latest RHEL 5 OS upgrade files are ADDM_OS_Upgrade_64_5.14.10.17_397381.ga.sh.gz (for the 64 bit architecture) and ADDM_OS_Upgrade_32_5.14.10.17_397381.ga.sh.gz (for the 32 bit architecture).

- **Content details and upgrade instructions**: To learn about the OS upgrade packages installed on your appliance after the latest OS upgrade, see the Latest RHEL 5 operating system upgrade and Latest RHEL 6 operating system upgrade pages. For upgrade instructions, see Upgrading the operating system. To learn about the earlier OS upgrade packages, from Previous operating system upgrades, select the link for the corresponding period.

The above-referenced product update packages are available for download from the BMC Electronic Product Distribution (EPD) site. Each package can be installed independently. See Accessing the release for more information.

### Accessing the release

This section provides the information that you need to search and download Technology Knowledge Updates (TKU) and Operation System Upgrades (OSU) from the BMC Electronic Product Distribution (EPD) website.

**To find and download TKU and OSU from the EPD website**

1. Create a directory in which to place the downloaded files.

    ![Note]
    
    On Windows computers, ensure that the directory is only one level into the directory structure. The EPD package creates a directory in the temporary directory when you extract the files, and the directory that contains the installation image should not be in a directory deeper than two levels into the directory structure.

3. Enter your user ID and password, and click **Submit**.
4. Click **Download Product**.
5. On the Export Compliance and Access Terms page, provide the required information, agree to the terms of the agreements, and click **Continue**.
6. If you are accessing this site for the first time, create an EPD profile on the Profiles tab; otherwise, skip to step 7:
   a. Under Localized Languages, select the language for the service pack.
   b. Under Install Platforms, select the platforms to download for the service pack.
   c. Click **Save Profile**.
7. Verify that the correct profile is displayed for your download purpose, and select the **Licensed Products** tab.
8. Find and select **BMC Atrium Discovery and Dependency Mapping** from the list of available products.
9. From the **Version** menu and the **Platform** menus, select the version and for which platform you want to download files.
10. Open the **Products** tab.
11. Select the check boxes next to the files and documents that you want to download.
   a. For the latest TKU, select the most recent **BMC Atrium Discovery and Dependency Mapping <version> Technology Knowledge Update (TKU) YYYY-MM-DD** file.
   b. For the latest OS upgrade, select the latest **BMC Atrium Discovery and Dependency Mapping - OS Upgrade - RHEL X - XX bits - Month Day Year** file. Ensure you select the proper RHEL version (RELH 5 or RELH 6) and the proper platform architecture (32-bit or 64-bit).
12. To download the selected items, click **Download (FTP)** or **Download Manager**:
   - **Download (FTP)** places the selected items in an FTP directory, and the credentials and FTP instructions are sent to you in an e-mail message.
   - **Download Manager** enables you to download multiple files consecutively, and to resume an interrupted download if the connection drops.
     This method requires a one-time installation of the AkamiNetSession Client program on the target computer and is usually the faster and more reliable way to transfer files. A checksum is used to verify file integrity automatically.

**Supported versions of BMC Atrium Discovery**

This page details the support status of existing versions of BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery). For complete definitions of release numbering, release terminology, and support levels, see the **BMC Software Product Support Policy**.

**Fully supported products**

The following table lists the fully supported product versions:
<table>
<thead>
<tr>
<th>Product</th>
<th>Release date</th>
<th>End of full support date</th>
<th>End of support date</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Atrium Discovery 10.1</td>
<td>November 14, 2014</td>
<td>November 30, 2017</td>
<td>November 30, 2019</td>
<td>BMC Atrium Discovery 10.1 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 10.0</td>
<td>March 17, 2014</td>
<td>April 30, 2017</td>
<td>April 30, 2019</td>
<td>BMC Atrium Discovery 10.0 documentation</td>
</tr>
</tbody>
</table>

**Version naming convention change**

The software support policy and versioning conventions used for BMC Enterprise Systems Management (ESM) software released as GA products after September 11, 2011. This date fell between the end of the 8.2 releases and the start of the 8.3 releases of BMC Atrium Discovery. Software products are now versioned as follows: $VV.RR.SP$ where $VV$=major version, $RR$=minor release, and $SP$=Service Pack.

**Limited support products**

This section lists the products which are in the limited support status:

<table>
<thead>
<tr>
<th>Product</th>
<th>Release date</th>
<th>End of full support date</th>
<th>End of support date</th>
<th>Documentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Atrium Discovery 9.0 SP3</td>
<td>July 2, 2014</td>
<td>November 30, 2015</td>
<td>November 30, 2017</td>
<td>BMC Atrium Discovery 9.0 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 9.0 SP2</td>
<td>September 6, 2013</td>
<td>November 30, 2015</td>
<td>November 30, 2017</td>
<td>BMC Atrium Discovery 9.0 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 9.0 SP1</td>
<td>March 14, 2013</td>
<td>November 30, 2015</td>
<td>November 30, 2017</td>
<td>BMC Atrium Discovery 9.0 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 8.3 SP3</td>
<td>November 16, 2012</td>
<td>November 30, 2014</td>
<td>November 30, 2016</td>
<td>BMC Atrium Discovery 8.3 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 8.3 SP2</td>
<td>February 27, 2012</td>
<td>November 30, 2014</td>
<td>November 30, 2016</td>
<td>BMC Atrium Discovery 8.3 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 8.3 SP1</td>
<td>December 14, 2011</td>
<td>November 30, 2014</td>
<td>November 30, 2016</td>
<td>BMC Atrium Discovery 8.3 documentation</td>
</tr>
<tr>
<td>BMC Atrium Discovery 8.3</td>
<td>October 12, 2011</td>
<td>November 30, 2014</td>
<td>November 30, 2016</td>
<td>BMC Atrium Discovery 8.3 documentation</td>
</tr>
</tbody>
</table>

When a product enters limited support:
New enhancements are not made to that version of the product. For reported issues, BMC Software Customer Support directs customers to existing fixes, patches and workarounds.

New Technology Knowledge Update (TKU) releases are not available for that version of the product.

During the time of limited support, BMC Software strongly encourages users of the respective versions of the product to upgrade to the latest current version.

### Upgrading to newer product version and migration path

The following table lists the migration path to upgrade to BMC Atrium Discovery 10.1 from your current version and the corresponding documentation links:

<table>
<thead>
<tr>
<th>Current version</th>
<th>Upgrade path</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Atrium Discovery 10.1.x</td>
<td>You can directly upgrade to 10.2.</td>
</tr>
<tr>
<td>BMC Atrium Discovery 10.0.x</td>
<td>You can directly upgrade to 10.1.</td>
</tr>
<tr>
<td>BMC Atrium Discovery 9.0.x on RHEL 6 (including 9.0 SP3)</td>
<td>You can directly upgrade to 10.1 .</td>
</tr>
</tbody>
</table>
| BMC Atrium Discovery 9.0.x on RHEL 5                 | You cannot directly upgrade to 10.2. Rather, you must do the following in the given order:  
  1. Back up your BMC Atrium Discovery data.  
  2. Restore the data on a 9.0.x on RHEL 6 appliance.  
  3. Upgrade to BMC Atrium Discovery version 10.1.  
  4. Upgrade to BMC Atrium Discovery version 10.2. |
| BMC Atrium Discovery 8.3.x                           | You cannot directly upgrade to 10.1. Rather, you must do the following in the given order:  
  1. Migrate your 8.3.x data to a new installation of BMC Atrium Discovery version 9.0 running on RHEL 6.  
  2. Upgrade to BMC Atrium Discovery version 10.1.  
  3. Upgrade to BMC Atrium Discovery version 10.2. |

### Most recent unsupported versions

The following table lists the most recent unsupported versions of BMC Atrium Discovery, requirements to upgrade to version 10.2 from these earlier version, and the corresponding documentation links:

<table>
<thead>
<tr>
<th>Current version</th>
<th>Migration path</th>
</tr>
</thead>
</table>
| BMC Atrium Discovery 8.2.x | You cannot directly upgrade to 10.2. Rather, you must do the following in the given order:  
  1. Upgrade to version 8.3.  
  2. Migrate your 8.3 data to a new installation of version 9.0.  
  3. Upgrade from 9.0 to 10.1 to 10.2. |
Limitations and restrictions of this version

Operational warnings

Warning

Failure to comply with the following instructions will result in datastore corruption, and in some cases unrecoverable datastore corruption.

Modification of datastore files and logs

Under no circumstances should you add, remove, or amend any of the datastore files or datastore log files without explicit clearance from BMC Customer Support.

The following are examples of datastore file names:

- pa55bc128f62ce9c427a1d742_nHost_hidx
- pa55bc128f62ce9c427a1d742_nHost_hist
- __db.001
- DB_CONFIG
- main

The following are examples of datastore log file names:

- log.000002301
- log.000002302

The location of the datastore and the datastore log files can be obtained by reading the /usr/tideway/etc/link.conf file. Under no circumstances should you modify this file once a system has been commissioned.

Datastore files must not be moved to a remote filesystem

The datastore files must not be moved to a remote filesystem. This is not a supported configuration, nor is this supported by the underlying Berkeley DB database environment.

Limitations and restrictions

The following list provides links to the limitations and restrictions of this version:

- Incorrect status in offline backup page
- SNMP discovery of Solaris 11 targets
- Appliance system disk larger than 2TB not supported
- Using WMI over IPv6
- SNMP discovery restrictions — AIX VIO, AIX 5.3, AIX 6.1
- SNMP discovery restrictions — AIX VIO
- SNMP discovery restrictions — Solaris 10
- SNMP discovery restrictions — Solaris 9
- Scanning without credentials often identifies hosts as an "Unsupported device"
- Hyper-V Windows virtual machines always reports NIC speed as 10 GBps
- Non-English Windows discovery using RemQuery
- Non-ASCII Unicode characters in CAM
- Tcpvcon cannot be pushed to Windows 2000 hosts (13963)
- Tcpvcon version later than 2.34 cannot return port information (QM001716854)
- Scanning a real host previously scanned using pool data (6079)
- Concurrent lock attempts can lock all users from editing the port scan settings
- Changes to user group memberships
- ECAError nodes show tracebacks of the error that occurred
- Third-party applications depending on Tideway security must be run after the security service has started
- NDD discovery interface support
- Computer CIs do not always reconcile correctly
- Record data should not be processed with tools that change line endings
- WMI might report incorrect memory
- WMI arguments might be truncated
- Home directory of discovery user on target computer must not be read-only
- Solaris 10 truncates process information for non-privileged users
- Solaris 8 and ifconfig
- Process information truncated in AIX
- OpenVMS support
- IP address change requires appliance restart
- "<attrib> = None" construct in WHERE clause not supported
- Processor type correctly reported only by non-srvinfo access methods
- Disabling "Ping hosts" setting slows Discovery
- AIX user password must be changed by user after creation by root
- SNMP credential does not validate IP address key
- Manual cron changes are overwritten
- Do not run service tideway status as root
- Search facility searches hidden attributes
- Modifying standard reports
- Visualization size limit (QM001711035)
- Xen para-virtualized hosts discovery limitations (QM001744086)
- IBM AIX machine serial number changes when moved from one host container to another (QM001775765)
- Internet Explorer 10 browser support limitations
- Discovering RAM in BMC Atrium Discovery 10.x using version 9.x Windows proxies
On such browsers, we have thoroughly tested the functionality and will consider it a bug if something does not work. These are the browsers we recommend using.

- Firefox 24.0 and above
- Chrome latest version
- Internet Explorer 8 (The Software Context view is not supported on Internet Explorer 8)
- Internet Explorer 9 (The Software Context view is not supported on Internet Explorer 9)
- Internet Explorer 10 (Desktop mode only, not Windows UI/Metro mode - see below)
- Internet Explorer 11 (Desktop mode only, not Windows UI/Metro mode - see below)

⚠️ Internet Explorer Compatibility Mode

Internet Explorer Compatibility Mode is not supported, as this forces the browser to behave as an older (unsupported) version. This behavior can be controlled with the Internet Explorer settings:

- Tools > Compatibility View
- Tools > Compatibility View Settings

Incorrect status in offline backup page

When performing operations with the Backup or Disk Configuration utilities, the user interface (UI) incorrectly displays the status to users logged in with a different ID as *The appliance has been shut down*, instead of showing that the operation is in progress. The problem occurs only when the browser is connected to the appliance through HTTPS, on an upgraded BMC Atrium Discovery 9.0 appliance running on RHEL 5.

**SNMP discovery of Solaris 11 targets**

Discovery of unpatched Solaris 11 systems using SNMP does not succeed. Applying Solaris 11 OS patches to update the SNMP agent to the current version should fix this.

**Appliance system disk larger than 2TB not supported**

The BMC Atrium Discovery appliance does not support installation on a system disk that is larger than 2TB, either by installation from the kickstart DVD or by extending the virtual disk for a virtual appliance. Disks larger than 2TB can be used for additional disks on the appliance.

**Using WMI over IPv6**

WMI over IPv6 is not supported on the following versions of Windows:

- Windows Server 2003
- Windows XP
- Windows 2000
  To discover these versions of Windows using WMI, you must use IPv4.
SNMP discovery restrictions — AIX VIO, AIX 5.3, AIX 6.1

- Some AIX-specific attributes are not set on DiscoveredNetworkInterface nodes (for example, interface_type, virtual_adapters, physical_adapters, shared_adapters and physical_location).
- Duplex and negotiation are not detected for network interfaces.
- Where IPv6 connections exist on a discovered host, DiscoveredNetworkConnections are created with illegal IPv4 addresses like 254.253.154.176, due to bugs in the AIX MIB.
- The following limitations result from the SNMP agent for these platforms only implementing older (deprecated) MIBs such as IP-MIB::ipAddrTable
  - IPv6 addresses, connections and listening ports are not discovered.
  - Processes are not associated with network connections or listening ports.

SNMP discovery restrictions — AIX VIO

- Fewer network interfaces might be discovered via SNMP than via a login credential.

SNMP discovery restrictions — Solaris 10

- Duplex and negotiation are not detected for network interfaces.
- Fewer IPv4 addresses might be discovered via SNMP than with a login credential.
- Discovery is not possible via IPv6 as there is no support for an IPv6 transport in the version of net-snmp provided with Solaris 10.
- The following limitations apply because the SNMP agent for these platforms only implements older (deprecated) MIBs such as IP-MIB::ipAddrTable.
  - IPv6 addresses, connections and listening ports are not discovered.
  - Processes are not associated with network connections or listening ports.

SNMP discovery restrictions — Solaris 9

- Duplex and negotiation are not detected for network interfaces.
- The following limitations apply because the SNMP agent for these platforms only implements older (deprecated) MIBs such as IP-MIB::ipAddrTable.
  - IPv6 addresses, connections and listening ports are not discovered.
  - Processes are not associated with network connections or listening ports.

Scanning without credentials often identifies hosts as an "Unsupported device"

Scanning without credentials often identifies hosts as an "Unsupported device". To obtain full information, scan using valid credentials.

Hyper-V Windows virtual machines always reports NIC speed as 10 GBps

Hyper-V Windows virtual machines always report NIC speed as 10 GBps regardless of the actual speed. Consequently, for Hyper-V Windows VMs, the host and switch mismatch field shows incorrect results.
Non-English Windows discovery using RemQuery

When attempting to discover Windows hosts using a non-English locale using only RemQuery, discovery may fail with the error message Failed to parse command output, or it may return incorrect information. They can however be successfully discovered using WMI.

Non-ASCII Unicode characters in CAM

In Collaborative Application Mapping (CAM), if you create components such as group names or functional components, that contain non-ASCII Unicode characters, the Business Application Instance (BAI) that results from running the pattern displays with unreadable characters.

Tcpvcon cannot be pushed to Windows 2000 hosts (13963)

Tcpvcon cannot be pushed to Windows 2000 hosts. The workaround is to deploy the utility manually. QM001683624

Tcpvcon version later than 2.34 cannot return port information (QM001716854)

To discover port information (getProcessToConnectionMapping) from computers running Windows 2000 or earlier, you must have version 2.34 of Tcpvcon installed on them. If a more recent version of Tcpvcon has been installed on the target, you must replace it with version 2.34 to discover port information.

Version 2.34 of Tcpvcon is shipped along with the BMC Atrium Discovery Windows proxies. To replace the recent version with version 2.34, perform the following steps:

1. On the computer running a Windows proxy, copy the tcpvcon.exe file, version 2.34, from the following location:
   \C:\Program Files\BMC Software\ADDM Proxy
2. On the target host, navigate to the location of the recent version of the tcpvcon.exe file and replace that file with the version 2.34 file that you have copied.
3. Run discovery again.

⚠️ If a recent version of Tcpvcon is installed on a remote host, execution of the Tcpvcon command on the BMC Atrium Discovery appliance will fail and display the following timeout error message in the Windows proxy debug logs of the host:

```
RemQuery(): user - TSL\admintest: Timed out status = FAILURE
```


The timeout error will be reported because recent versions of Tcpvcon require a GUI-based end-user license agreement (EULA) to be confirmed when it is run for the first time. If you confirm the EULA on the host either manually or by using the accepteula switch, the Tcpvcon command is invoked successfully. However, as BMC Atrium Discovery does not support recent versions of Tcpvcon, parsing of the command output will fail and the following error message will be displayed in the log:

Failed to parse command output status = FAILURE

### Scanning a real host previously scanned using pool data (6079)

When you upload scanner files to the appliance and run it in playback mode, .no-expiry files are created for each IP address. This means that this pool data will not be deleted at the next scan, and subsequent discovery runs will operate by playing back the pool data rather than by scanning the real IP address. This is true if the appliance is operating in Record or Playback mode. If you subsequently attempt to scan the real IP address, the pool data will not be updated if the .no-expiry file is present. If you are scanning an IP address and it is not being updated, you should check the pool data for existence of a .no-expiry file and delete it. The pool data structure is:

```
/var/pool/xx/xx/xx/xx/.no-expiry
/var/pool/xx/xx/xx/xx/<data>
```

where /xx/xx/xx/xx is the IP address of the host.

For more information on scanning hosts from scanner files and how to handle pool data, see [Standalone UNIX scanning](#).

### Concurrent lock attempts can lock all users from editing the port scan settings

In the port scanning page, if a user locks it for editing and another user subsequently tries lock it, the second user's attempt fails. If the user who successfully locked the page cancels the operation and leaves the page, it remains locked for the unsuccessful user, and on refresh for the successful user too.

### Changes to user group memberships

If the privileges of a BMC Atrium Discovery user are extended by changing the user's group memberships, then these changes might not take effect for up to 5 minutes. However if privileges are withdrawn from the user these changes take immediate effect.
ECAError nodes show tracebacks of the error that occurred
This could cause concern during ethical hacking tests but is not actually a problem because the code shown is from patterns, which are already visible to the user, not internal to the product.

Third-party applications depending on Tideway security must be run after the security service has started
If third-party applications that depend on the Tideway Security Service are run before it has completed initialization, they will fail as you cannot validate permissions and users from the Security Service.
Ensure that the Tideway Security service has completed initializing before running third-party applications.

NDD discovery interface support
NDD discovery does not support trp interfaces.

Computer CIs do not always reconcile correctly
On certain UNIX systems, BMC Atrium CMDB cannot reconcile the same Computer System CIs from the BMC Performance Management (BPM) and BMC Atrium Discovery datasets. If the <hostname> command is not configured correctly on these systems, the command returns the fully qualified domain hostname (FQDN) instead of just the host name, resulting in duplicate Computer System CIs in the BMC.ASSET dataset.

Possible solutions include correcting the command output on the affected system, or disabling reconciliation with data from BPM. If you are unsure, contact your Customer Support representative to discuss additional options.

Record data should not be processed with tools that change line endings
BMC Atrium Discovery stores record data in UNIX and DOS formats. UNIX format files have LF line endings, and DOS format file have CR LF line endings. If you process the record data with a tool that changes line endings, you will see exceptions in the Discovery logs.

WMI might report incorrect memory
WMI might report the physical memory available on Windows hosts incorrectly.

WMI arguments might be truncated
In unusual situations the first argument to a process might not be reported to discovery by the target Windows host. This happens when a Windows process was created with CreateProcess with the ApplicationName parameter specified but without the module name used as the first argument passed in the CommandLine parameter.

Home directory of discovery user on target computer must not be read-only
The home directory of the user that is used for discovery on target hosts must not be read-only. If it is read-only, scripts (such as which on Solaris 9 and 10 hosts) that write to the home directory will fail.
Solaris 10 truncates process information for non-privileged users

In Solaris 10 /usr/ucb/ps will now only output the first 79 characters of commands unless it is run as root. The reason for this change is to prevent the inadvertent leak of private process data. Where process information is truncated, Discovery will be incomplete for that host. You must add the proc_owner right for the user account used for discovery, for example, the tideway user. To do this and retain all of the default privileges, as root, enter:

```
usermod -K defaultpriv=file_link_any,proc_info,proc_session,proc_fork,
      proc_exec,proc_owner tideway
```

No spaces are permitted in the defaultpriv argument.

Solaris 8 and 9

Patches have been rolled out to replicate this behavior on Solaris 8 and 9.

- Solaris 8 patch — 109023-05
- Solaris 9 patch — 120240-01

To workaround this, you should deploy sudoers privileges for /usr/ucb/ps.

Solaris 8 and ifconfig

In Solaris 8 there are two ifconfig binaries:

- /sbin/ifconfig
- /usr/sbin/ifconfig

In all versions of Solaris other than 8, there is a single binary and a symbolic link.

The default path statement set by BMC Atrium Discovery ensures that /sbin/ifconfig is called first. In Solaris 8 this is the incorrect version, /usr/sbin/ifconfig must be run to obtain the correct information. To ensure this is the case, edit the ifconfig discovery script to specify the full path to ifconfig:

```
IFCONFIG=`PRIV_IFCONFIG /usr/sbin/ifconfig -a 2>/dev/null`
echo "$IFCONFIG"
```

Do not modify the path statement to correct this issue as that will cause other problems.
Process information truncated in AIX

On AIX the `ps` command limits output to the horizontal screen size. This can be overridden using the COLUMNS environment variable, though the maximum permissible value for this is 2047. Piping the output of the `ps` command through `cat` removes the columns restriction on AIX hosts with a May 2007 Service Pack.

OpenVMS support

Support for OpenVMS is limited to systems running the native vendor TCP stack.

IP address change requires appliance restart

Where the IP address of the appliance is changed, for example, by DHCP or a manual change, the appliance must be restarted.

"<attrib> = None" construct in WHERE clause not supported

The Search Service uses a number of Python constructs. However, the "= None" construct should not be used to recognize undefined values. For example, the following query does not work and returns nothing:

```plaintext
SEARCH SoftwareInstance
    WHERE version = None
```

You should state explicitly that you are looking for an undefined attribute. For example:

```plaintext
SEARCH SoftwareInstance
    WHERE name HAS SUBWORD "Web"
    AND NOT version IS DEFINED
and
SEARCH SoftwareInstance
    WHERE NOT version IS DEFINED
```

Processor type correctly reported only by non-srvinfo access methods

Processor type is correctly reported when using WMI and non-srvinfo Discovery methods. However if you discover the same host with srvinfo then it is reported incorrectly. Ensure that the WMI or non-srvinfo access method is enabled.

Disabling "Ping hosts" setting slows Discovery

If you disable the "Ping hosts before scanning" setting in the Discovery Configuration page, Discovery will try a number of methods before determining that there is no device at that IP address. If pinging is enabled, Discovery determines that there is no device immediately.
AIX user password must be changed by user after creation by root

On AIX, when a user password is changed by the root user, that password must be changed by the user at the next log in. If the password is not changed and Discovery is attempted using that user name and password, it fails when prompted to change the password. To prevent this from happening, if you are the root user and add a new user, log in as that user and change the password.

SNMP credential does not validate IP address key

When adding or editing a login or SNMP credential, the IP Address key does not validate the format. You are permitted to enter special characters, alphanumeric, and invalid IP address formats (172.17.1.3.3.4). Only enter valid IP addresses.

Manual cron changes are overwritten

If a cron job is manually edited this will not be noticed, and any change will be silently thrown away. This could be an issue where a manual change is made by someone not realising there is a cron management process.

The script should be scheduled using the cron feature (in $TIDEWAY/etc/cron/) as the tideway user.

Do not run service tideway status as root

Running the service tideway status command as root will cause the ownership of the database log files to change to root. Eventually this will cause the system to fail to start.

Always run sudo service tideway status as the tideway user, never as root. That is, sudo /sbin/service tideway status.

Search facility searches hidden attributes

The search facility searches hidden attributes and system fields, even though the users cannot normally see this information.

This was observed when searching for a subnet to add relationship to from a host. The search string 127 was entered and the following two subnets were returned:

- 192.168.115.0/24
- 172.16.203.0/24

The search string does not appear in the subnets, but might have been found in hidden attributes associated with the subnets. This behavior can be confusing.

Modifying standard reports

If you place an updated reports.xml file on a system without stopping the tideway services, you might see a traceback in the UI. To avoid this stop the tideway services before adding a new or modifying the existing reports.xml file.
Visualization size limit (QM001711035)
There is a visualization size limit of 5631x4439 pixels. Larger visualizations could consume excessive system resources and cause instability.

Xen para-virtualized hosts discovery limitations (QM001744086)
BMC Atrium Discovery has the following known limitations in discovering the Xen para-virtualized hosts:

- The hosts are not discovered as virtual.
- The corresponding UUID is not correctly discovered.
- The relationship between the Xen server and the virtual machine containers (software instances), including the virtualized hosts running on it, is not discovered.

There are no workarounds to overcome this limitation.

IBM AIX machine serial number changes when moved from one host container to another (QM001775765)
When an IBM AIX machine is moved from one host container to another, the discovered serial number of the host changes. As a result, a new host node is created and synchronized to the BMC Atrium CMDB (if CMDB synchronization is configured). The earlier host node is automatically removed through aging based on the Model Maintenance settings.

It is possible to manually destroy the earlier host node before it is removed through aging. However, manual destruction of host nodes is not recommended in production appliances. For more information, see Destroying data.

Internet Explorer 10 browser support limitations
BMC Atrium Discovery does not yet have a comprehensive level of browser support for Internet Explorer 10. However, 9.0 SP2 and later versions support Internet Explorer 10 at an ad-hoc level with the following known issues:

- When a dialog box is displayed (for example, the Add a New Run dialog), the background appears black.
- For creating conditions for functional component definitions, if you attempt to use named values, the drop down menus for selecting the operator conditions do not display any value.

In addition to these known issues, there might be other issues which have not yet been detected. Currently, there are no workarounds to the known issues and BMC recommends that you use a browser with a comprehensive level of support.
Discovering RAM in BMC Atrium Discovery 10.x using version 9.x Windows proxies

When BMC Atrium Discovery 10.0 uses version 9.x proxies to discover Windows hosts, you get information about either logical RAM or physical RAM, depending on the successful scanning method. It is recommended to use BMC Atrium Discovery 10.x proxies, as you might discover logical and physical RAM, when scanned using WMI.
Getting started

The Getting Started topics are intended for anyone who is starting to evaluate, use, or deploy BMC Atrium Discovery, including the BMC Atrium Discovery Community Edition. This section covers BMC Atrium Discovery planning, installation and configuration, guides you through core procedures for getting quick results from your discovery process, and suggests the next steps you might take to improve your discovery results.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Related information</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn the basics of the BMC Atrium Discovery product.</td>
<td>Key concepts</td>
<td>Understand key concepts and core functionality so that you can begin using BMC Atrium Discovery right away.</td>
</tr>
<tr>
<td>Start using BMC Atrium Discovery.</td>
<td>End to end process</td>
<td>Run your first basic discovery without credentials, examine the results, and configure credentials for complete discovery. Schedule discovery runs and prevent discovery of specified IP addresses or ranges.</td>
</tr>
<tr>
<td>Learn more about the capabilities of BMC Atrium Discovery.</td>
<td>Next steps</td>
<td>Learn about the way that you can further enhance the discovery of your IT infrastructure and business applications such as patterns and TKUs, deep database discovery, and Collaborative Application Mapping.</td>
</tr>
</tbody>
</table>

Key concepts

This section contains the following information important to understanding BMC Atrium Discovery:

- Business Service Management
- Big Discovery
- BMC Atrium Discovery
- Glossary

Business Service Management

Business Service Management (BSM) is a comprehensive approach and unified platform for running IT. BSM offers a way to bring together many disparate processes and tools to create quantifiable improvement in efficiency and the ability to view technology as it applies to business process. BSM enables your IT department to:

- Operate by service rather than by individual configuration items or technology.
- Prioritize your efforts, improving the service that you deliver to your business.
- Understand and predict how technology impacts your business and your business impacts the IT infrastructure.
This diagram illustrates the aspects of Business Service Management based on ITIL processes.

BMC Software delivers a BSM solution according to the key functions of an IT organization:

- **Planning** — Capture and track information related to demand, supply, resources, finances, and risk.
- **Support** — Reduce complexity, integrate customer support, change management, and asset management, and increase their efficiency.
- **Operations** — Find and fix problems before your users do with analytics, event automation, and performance management. Automate repetitive, manual tasks to eliminate errors and get things done more quickly.

The IT Infrastructure Library® (ITIL®) defines standards for the processes that IT departments use to manage IT hardware and software, such as problem management, incident management, and change management. These processes ensure the availability of the critical IT services that sustain a business, like banking systems, ordering systems, and manufacturing systems.

Additionally, ITIL configuration management needs reliable data about components in the IT environment and needs to understand the impact to key business services when there are changes to that environment. For a large company, maintaining information about thousands of pieces of hardware and software is difficult to manage. Central to these ITIL processes is the Configuration Management Database (CMDB). The data in the CMDB feeds the related applications that perform ITIL processes.

As an enabler to ITIL Configuration Management, BMC Atrium Discovery automates the process of populating CMDB by exploring IT systems to identify hardware and software, and then creating CIs and relationships from the discovered data. With BMC Atrium Discovery, the Configuration Manager is able to
• Dramatically decrease CMDB population times, which accelerates the implementation of BSM and the applications that perform ITIL Service Operations processes.
• Increase data accuracy by automatically and continuously updating configuration data in the CMDB.
• Identify the IT components that comprise enterprise applications and business services.

**Big Discovery**

**The problem**

Today, IT teams who are responsible for managing and maintaining the configuration state of the largest IT estates are struggling with obtaining an accurate picture of their entire environment. Configuration Managers are also dealing with the challenge of highly distributed environments that they must consolidate into a holistic view, in order to make the best decisions for their business.

• **More dynamic data centers** — Given the rate of change in the data center, users need near real-time data. Their goal is daily discovery scans and CMDB updates.
• **Time constraints for access to the network** — Many users have limited windows during which they can scan the network. Often several days are needed to fully scan their data center.
• **Tool constraints for discovery** — The tools available that are meant to solve these challenges today are too slow when scanning and compiling data into usable results and require multiple days to do so. Often, in the time it takes this data to be converted into something useful, it can already be outdated. Additionally, the solutions add to the difficulty of managing large scale environments by introducing maintenance and management needs of the tools themselves and high learning curves with cumbersome UIs.
• **Data Currency for more dynamic data centers** — Despite the rate of change in the data center, users continue to need near real-time data. Due to this pace, Configuration Managers are unable to refresh data center inventory and dependency information at the desired rate.

These challenges IT face are as massive as the organizations they manage. They do the best they can, with the tools they have but they want better, faster, and smarter.

**The solution**

The ideal solution delivers a single, consolidated view of a very large data center (in excess of 100K OSIs) more rapidly than before. All this with a tool that requires low administration and with a smart UI that keeps things easy.

**BMC Atrium Discovery version 10.x Big Discovery**

Big Discovery - Discover and get usable results from even the largest Data Centers in the shortest time possible.

• Leverages new BMC Atrium Discovery appliance clustering technology
• No Scanning limits – just add hardware
- Fault tolerance designed to manage hardware failures
- Highly Scalable End User UI
- Transparent cluster management

BMC Atrium Discovery version 10.x Big Discovery allows an IT Organization to always understand the ever changing state of the smallest to the largest enterprise. BMC overcomes modern challenges of extreme scale by leveraging its new clustering technology — Big Discovery. Thanks to this, IT can now discover and produce near real time intelligence about the data center to support business critical decisions. Already the fastest discovery solution — now even faster with BMC Atrium Discovery version 10.x Big Discovery!

For more information on the details of Big Discovery, see the following:

- Cluster overview
- Fault tolerance
- Creating a cluster
- Adding a machine to an existing cluster
- Changing the machine that is the coordinator
- Removing a machine from a cluster
- Changing the address of a machine
- Reverting a cluster member into a standalone machine
- Shutting down and restarting machines in a cluster
- Shutting down and restarting clusters

**BMC Atrium Discovery**

BMC Atrium Discovery and Dependency Mapping is a data center discovery solution that automatically discovers data center inventory, configuration and relationship data, and maps applications to the IT infrastructure. BMC Atrium Discovery establishes the foundation for improving IT processes and productivity by providing timely and actionable insight to make informed decisions in IT service management, asset management and infrastructure/operations management. Benefits include reducing the number of incidents caused by change, reducing the time taken to isolate the root cause, reducing the time taken to prepare for audits, preventing audit penalties, prioritizing incidents based on business impact, and optimizing data center hardware and software.

- **Find It** — Guarantee data accuracy and reduce costs to collect data center inventory, configuration and relationship data via an automated discovery process.
- **Manage It** — Leverage rich data to improve IT processes and productivity by replacing guesswork with data-driven decisions.
- **Optimize It** — Add business context to IT management processes by understanding how the data center infrastructure supports business applications.

**Virtual Appliance with Agentless Discovery**

Usable information within hours:
BMC Atrium Discovery is available as a ready-to run appliance. Deployment takes literally minutes – there is no installation process to run through, no external database to install and configure, no specific configuration and operating system requirements to satisfy. This allows you to get up and running very quickly.

With an agentless approach there is no software to roll out to servers; therefore it is the best solution that achieves a balance between discovery depth, impact to the infrastructure, and administration overhead.

Data Provenance

Data accuracy you can verify and trust. For an automated discovery tool to be trusted and accepted by the user community, it must provide transparency into how the data was obtained. BMC Atrium Discovery’s Provenance feature shows the actual command that was executed, the output of the command, and the timestamp when it happened. There’s no searching through log files – all this information is available right in the UI.

Provenance provides indisputable evidence why this data can be trusted and thus speeds adoption of the data into IT processes.

Broad Pattern Library; Simple to Extend

Tailor the discovery process to fit your organization.

The BMC Atrium Discovery library includes over 650 patterns for industry leading software such as Oracle, SAP, IBM, HP, VMware, Hadoop, Citrix and more. New patterns are added monthly.

With BMC Atrium Discovery, it’s easy to fine-tune and extend the discovery process to meet the needs of your organization. Discover additional attributes, custom software, uncommon SNMP devices and more. Simply modify a discovery pattern or use one of the supplied pattern templates. All patterns are text-based and extending a pattern is as easy as modifying a script or batch file.

The following diagram illustrates the relationships between the different parts of the BMC Atrium Discovery system:
This diagram illustrates the relationship between the different elements of BMC Atrium Discovery.

**BMC Atrium Discovery - what it does**

BMC Atrium Discovery and Dependency Mapping automatically discovers the hardware and software in your data center, determines configuration and relationship data, and maps applications to the IT infrastructure.

- Discovery
- Reasoning
- Datastore
- CMDB Sync
- Collaborative Application Mapping (CAM)
- Big Discovery

**Discovery**

Information about your organization's hardware and software is obtained by the Discovery Engine. It uses credentials, which are held in a secure encrypted vault, to log into target hosts using variety of methods, such as SSH, Telnet, WMI and SNMP. Once logged in, the Discovery Engine executes commands, the results of which are returned and stored in the datastore as directly discovered data (DDD). Windows discovery uses an external proxy running on a dedicated windows server to log into and scan Windows hosts in the estate.

**Reasoning**

The Discovery Engine is supported by Reasoning which intelligently infers information about hosts and programs from the DDD returned. The process of adding DDD to the datastore causes Reasoning to execute patterns against the DDD. Each pattern represents knowledge about particular software or hardware and Reasoning uses this knowledge to create more detailed
"inferred" data. Inferred data is the representation of the scanned IT environment and is stored in the datastore. The provenance of each item of inferred data is also stored meaning that when examining an inferred entity in the UI, you can examine the information which was used to create it.

Patterns can be updated, either through monthly Technology Knowledge Update (TKU) releases, or by writing new patterns using The Pattern Language (TPL).

**Datastore**

The datastore is the database in which the DDD and inferred data is stored. Data written to the datastore is instantly indexed allowing you to search for items of interest using simple keywords in the UI. In addition to the DDD and inferred data mentioned, the datastore holds TKU information, patterns, operational data and some configuration data.

The datastore uses a graph model meaning that it represents data as nodes connected to each other with relationships. This is more suited to modeling the complex relationships in an IT environment than a relational database.

**CMDB Sync**

CMDB synchronization provides a means of keeping data in the BMC Atrium CMDB continuously synchronized with information discovered by BMC Atrium Discovery. The BMC Atrium Discovery data model is different from the Common Data Model (CDM) used in the CMDB, so the synchronization mechanism transforms the required information from one data model to the other.

**Collaborative Application Mapping (CAM)**

Collaborative application mapping simplifies and automates the creation of self-maintaining application maps by providing a workflow and collaboration tools that make it easy to obtain information about applications from the application owner. A dynamic, automatically maintained, effective application map enables you to understand the key relationships between how your business operates and the infrastructure that supports it. It also becomes the initial, crucial part of Service Impact Analysis by maintaining accurate service models for BSM.

Collaborative application mapping ensures that any time there are changes in the IT environment, BMC Atrium Discovery will discover those changes automatically so that the application map always reflects the current environment. And most important, this self-maintaining application map is automatically generated without the need for programming.

**Big Discovery**

Regardless of data center size or complexity, Atrium Discovery delivers a refreshed view of your data center as often as needed.
Big Discovery delivers powerful, actionable data center insight in the shortest possible time:
• Leverages new appliance clustering technology
• No scanning limits – just add hardware
• Transparent cluster management
• Fault tolerance to manage hardware failures
• Highly scalable end-user UI

Big Discovery enables you to discover and get usable results from even the largest Data Centers in the shortest time possible. Big Discovery uses clusters to achieve this. A cluster consists of two or more coordinated BMC Atrium Discovery machines, one of which is in control of the group and is referred to as the coordinator. Additional machines can be added to the cluster at any time. When you do so, the system spreads the existing data out amongst all the machines evenly, in order to maximize performance and utilization of each machine in the cluster. This can take a while to happen, but the system is fully usable while it happens: you can do discovery and browse the data.

Clusters can be configured with fault tolerance meaning that if a machine fails, it can be removed and replaced, without any consequent loss of data, and without interrupting normal operation.

Managing your cluster has been made as simple as possible. Use any machine in the cluster to manage all others: update TKUs, add credentials and scans, and even upgrade the product itself from a single place.

**Glossary**

This section provides a glossary of terms.

**Active Directory Windows proxy**
A BMC Atrium Discovery application that runs on a customer-provided Windows host that is part of an Active Directory domain or workgroup. The user that the discovery service runs as is configured after the Windows proxy is installed. Where that user is configured on hosts in the domain, the Windows proxy can log in and run discovery commands. The Active Directory Windows proxy does not use any credentials that are entered using the BMC Atrium Discovery user interface.

**application map**
A dynamic, automatically maintained representation of application structures in your environment. An effective application map identifies the key relationships between how your business operates and the infrastructure that supports it. It also becomes the initial, crucial part of Service Impact Analysis by maintaining accurate service models for BSM.

**blackout window**
A configuration that enables you to prevent access to the CMDB during sensitive times. During a blackout window, all processing of the CMDB synchronization queue is paused, and processing cannot resume until the window ends.
BMC Atrium Discovery

Automates the process of populating the BMC Atrium Configuration Management Database (BMC Atrium CMDB) by exploring IT systems to identify hardware and software, and then creating configuration items (CIs) and relationships from the discovered data.

bonding

Enables you to join two NICs as a single physical device so that they appear as one interface. This is usually performed to provide failover capabilities or load balancing. Also known as teaming.

Collaborative Application Mapping

The process of investigating and understanding the applications in an estate, and creating the model that represents the application in the datastore.

command-line utility

A tool that you can run on a command-line interface to configure BMC Atrium Discovery by obtaining information from specific systems.

component

A general term that is used to mean one part of something more complex. For example, a computer system might be a component of an IT service, and an application might be a component in an application server.

consolidation

The playback of scanned data from multiple scanning appliances to a single consolidation appliance.

Credential Windows proxy

A BMC Atrium Discovery application that runs on a customer-provided Windows host and uses credentials supplied by the BMC Atrium Discovery appliance to perform Windows discovery.

data aging

Discovered data is regarded as valid at the time of its last successful scan. The nature of IT infrastructure means that frequent, minor changes to configurations, hosts, and software are common. Consequently, discovered data can be regarded as becoming less current with the passing of time. In BMC Atrium Discovery when data passes a certain configurable aging threshold, it is destroyed.

Data Integration Point (DIP)

Code that handles communication with target systems. BMC Atrium Discovery connects to target systems in defined ways (such as through JDBC drivers, a protocol, or specific credentials), and that connection is determined by the DIP.

datastore

All data used by the BMC Atrium Discovery system is held in an object database. The datastore treats data as a set of objects and the relationships between them.
Directly Discovered Data
Data that the Discovery Engine has discovered that has been parsed but not processed. Although the information is not yet classified or understood, it is stored in a structured form that can be queried and reported on, making it easy to construct certain kinds of discovery reports and to develop new patterns.

development appliance (on the customer site)
The appliance on the customer site to test the custom patterns and other customer-specific configurations prior to deploying changes to BMC Atrium Discovery on the production environment. The changes are deployed in the customer's production environment only after verifying and confirming that the development appliance is stable and functions as intended.

Discovery
The part of the BMC Atrium Discovery system that communicates with host systems, and obtains information from them. Discovery is driven by Reasoning which infers detailed information about hosts and programs and populates the datastore. See also Reasoning Engine.

Discovery endpoint
The endpoint of a single Discovery access, the IP address of the discovery target.

Discovery Run
A scan of one or more Discovery endpoints, specified as an IP address, address, or range of addresses that are scanned as an entity. For each Discovery Run, a node is created that records information such as the user who started the run, the start and end time, and so forth.

event
A change or action that affects the discovery process, such as a software instance that was created or updated. In BMC Atrium Discovery, the Rules Engine (ECA Engine) executes rules in response to events.

functional component
A node created by patterns based on Functional Component Definitions. The functional component is a single block of information that combines similar functionality into logical groups that help application owners and data consumers discover applications at discovery time.

Functional Component Definition
The specification that is used to create a functional component. Functional Component Definitions (FCDs) help application owners and data consumers define and develop application structures. The primary goal of an FCD is to help provide tangible data to BMC Atrium Discovery during an iterative and collaborative application modeling process to build an appropriate application model.
host
A node in the model which represents a physical or virtual computer system including information about its OS and its physical or virtual hardware. A host is sometimes referred to as an OSI (Operating System Instance). See Glossary.

ID
A unique identifier for a node (also known as a node ID). For a BMC Atrium Discovery node, an internal identifier that is used as an index by the database. It is a binary identifier represented in hexadecimal format. An ID is not intended to be human-readable; it is designed to be used by the datastore (for example, 4e4fd2c2ae4ccf123272d8446e486f7374. It identifies a stored node, not the item that the node represents. If the node corresponding to an entity is destroyed, and a new node is subsequently created for it, the new node will have a different ID, but it will have the same key.

inferencing
The act of drawing conclusions about data based on what is known about other data.

key
A unique identifier for the entity that a node represents. Unlike the node ID, the key of a node is persistent.

kind
The type of a node, such as Host, Application Version or Person. Also referred to as node kind.

lifecycle
The conditions that describe when an entity comes into existence to when it no longer exists. For nodes in the BMC Atrium Discovery model, the lifecycle stages are:

- **Current** — Describes nodes that exist in the model. BMC Atrium Discovery contains evidence that nodes currently exist in your environment.
- **Aging** — Describes nodes that exist in your model; however, they represent entities that BMC Atrium Discovery has not detected in a certain period of time, and has 'aged out' of the model. BMC Atrium Discovery does not always age entities that it cannot identify over a period of time.
- **Destroyed** — Describes nodes that have been marked as destroyed (yet remain in the model).
- **Purged** — Describes destroyed nodes that have been purged from the model. Purging a node indicates that it no longer exists in the model and it has been removed from the datastore.

logical host
A hardware or software host that is contained in a virtual machine (software), a collaborating host in a cluster (hardware) or a blade in a blade server (hardware).
node
An object in the BMC Atrium Discovery datastore that represents an entity in the environment. Nodes have a kind, such as 'Host', and a number of named attributes. Nodes can be connected to other nodes using relationships. Most node kinds have a key that uniquely identifies the entity in the environment.

node ID
See ID.

node kind
The type of a node, such as a Host or Software Instance. The default set of nodes and their associated attributes and relationships are defined in the BMC Atrium Discovery taxonomy.

pattern
In BMC Atrium Discovery, the Pattern Language (TPL) that creates and maintains the model. Each pattern in TPL has a corresponding pattern node in the model, which is related to the nodes that the pattern maintains. Patterns are used to extend the functionality of the reasoning engine.

production hours
The core business hours corresponding to a specific time zone. Typically, Discovery scans are scheduled at non-production hours to avoid any potential impact on the BMC Atrium Discovery end users, or target critical systems when they are the busiest, or schedule any CMDB synchronization black-out windows to avoid impacting the AR System and CMDB end users.

provenance
Meta-information describing how inferred information came to exist. It is generated as Reasoning builds and maintains the model. Provenance information is stored as relationships in the model.

Reasoning Engine
An event-based engine that orchestrates and drives the population of different parts of the data model through a series of rules that make up the core functionality of the BMC Atrium Discovery product. It is extensible through the use of patterns.

relationship
The way that objects are associated with each other. Relationships are non-directional, and are defined by the roles represented by each object. They are stored in the datastore in the format Node:Role:RelationshipLink:Role:Node.

relationship link
The connection between two roles in a relationship.

RemQuery
A utility that enables you to execute commands on remote Windows hosts in a similar way to the commercial PsExec utility.
When BMC Atrium Discovery requests a discovery action using the RemQuery utility, RemQuery copies a binary (itself) to the $ADMIN$ share on the target system, and then installs and runs that binary as a service. Each of these steps requires Local Administrator permissions. The service is then used to execute the discovery scripts. At the end of the scan, the service is stopped and uninstalled, but the executable is left in the $ADMIN$ share. If a copy already exists, it is not copied again.

removal
The concept of taking data out of the model using one or more of the BMC Atrium Discovery lifecycle methodologies (Aging, Destroyed or Purged).

role
The responsibility or actions of the relationship between two nodes. A node with a relationship to another node acts in a role in the relationship, which indicates its part of the relationship. For example, in a 'Dependency' relationship, one node has the role 'Dependant' and the other has the role 'DependedUpon'.

Rules Engine
Another term used to describe the Reasoning Engine. The Rules Engine processes the rules that are generated from Patterns, in order to maintain the model. The Rules Engine is an Event Condition Action (ECA) engine.

rules
Small fragments of executable code that run in the Rules Engine in BMC Atrium Discovery. Rules are generated from patterns when they are activated. Additional core rules are distributed with BMC Atrium Discovery.

scanner file
A Error rendering macro 'excerpt-include' : No link could be created for 'Using Scanner Files'.

seed data
In Collaborative Application Mapping, a small sample of host names that are involved in the application or component names. The goal of finding seed data is to provide just a few pieces of information to the application owner (typically communicated through e-mail or instant message) that are clues to help determine what to start investigating.

session establishment duration
The time it takes to establish a session to log in to the host. See also Total Duration and Total Discovery Duration.

taxonomy
The template defining the nodes, attributes, and relationships used by BMC Atrium Discovery and stored in the datastore. The BMC Atrium Discovery taxonomy also defines how much of the data model is represented in the user interface.
total discovery duration
The time it takes to establish a session and to run commands. See also Session Establishment Duration and Total Duration.

total duration
The time it takes to discover and process the data from the target (the duration between the start and end times). See also Session Establishment Duration and Total Discovery Duration.

trigger
The conditions under which a pattern executes. Triggers correspond to the creation, confirmation, modification, or destruction of a node.

Windows proxy
A discovery proxy that is installed on a Windows system, on which the discovery process is controlled by a Linux-based appliance (known as the master). See Active Directory Windows proxy and Credential Windows proxy.

End to end process
This section contains an overview of the following topics:

- Understanding BMC Atrium Discovery
- Using BMC Atrium Discovery

Understanding BMC Atrium Discovery
This section is intended for IT operations staff and managers who are new to BMC Atrium Discovery and need an initial orientation. It gives a brief overview of the following topics:

- Goals and benefits
- The discovery process

Goals and benefits
The ultimate goal of BMC Atrium Discovery is to identify systems in the network and obtain relevant information from them as quickly as possible and with the lowest impact, using a variety of different tools and techniques to communicate.

Use of BMC Atrium Discovery in your organization gives you an insight into the elements that build up your IT infrastructure. This is helpful when you are following the ITIL best practices and enables you to keep the configuration and asset database up to date as well as verify compliance of your network, hardware and/or software configuration to the uniform organization-wide golden standard.

For additional terms and definitions, see the Glossary.
The discovery process

BMC Atrium Discovery attempts to discover whatever device is at each IP address or endpoint. For each endpoint, it follows the following procedure. The first check is to see whether it is permitted to scan the endpoint. If not, no further action is taken. If scanning is permitted, discovery checks for cached data for that IP address. If no cached data is found, then it checks for a previously successful login method, for example, an ssh login, Windows Proxy, or SNMP access, and uses that method to log in again and run discovery commands. If none of this is successful, BMC Atrium Discovery performs an access method port scan. If there is no response, the IP address is considered dark space and no further discovery is attempted. For an in-depth description of dark space scanning, see Dark space scanning.

If there is a response, that is, there is a device at that endpoint, BMC Atrium Discovery attempts the following access methods, in sequence until one is successful:

- Attempt to login (using shell first, and then using Windows proxy if shell is unsuccessful) and run discovery commands if suitable credentials or an Active Directory Windows proxy are available.
- Perform SNMP get.
- Connect using telnet to read the banner.
- Connect to the z/OS Host Server port.
- Perform an HTTP HEAD request on the host.
- Connect using ftp to read the banner.
- Match open ports (IP fingerprint) to predict a class of OS.

BMC Atrium Discovery is likely to be able to identify the OS and version without requiring all of the steps described previously. After it has discovered sufficient information, the discovery service stops working on that IP address and moves on to the next address in the queue.

Any device that BMC Atrium Discovery cannot log into is identified only by the results obtained from reverse DNS lookup, telnet, SNMP requests, and IP fingerprinting (where enabled). Hosts and mainframe computer nodes are created only after a successful login. Network device nodes are created after a successful SNMP access.

⚠️ UTF-8 Data

BMC Atrium Discovery stores and supports all character data as UTF-8, but does not support direct discovery of data outside the basic ASCII character set. If such data is discovered, BMC Atrium Discovery is unable to map it to UTF-8 and invalid characters are introduced into the discovery data. Windows proxies do attempt some conversion of this data.
The Community Edition

The Community Edition of BMC Atrium Discovery is a version of BMC Atrium Discovery which is available for download. The community edition can be used to scan up to 60 devices in your production environment. When more than 60 inferred nodes have been created in the data store, you cannot create any new discovery runs and discovery is disabled. This prevents any accidental breach of the license agreement.

You are periodically reminded of the terms of this agreement by a pop up dialog. The dialog can be dismissed by clicking the Close button.

The dialog provides the following links:

- Forums - opens the Community Forums.
- Leave Feedback - opens an online survey which gives you the opportunity to provide feedback on BMC Atrium Discovery Community Edition.
- Terms and Conditions - opens a page which shows the license terms of the Community Edition.

Additional limitations

In addition to the limitation on discovery, the following features are disabled:

- Clustering
- Multi-Appliance Consolidation
- CMDB Sync
- Integration SDKs and Export CSV, XML
- Productised Integrations
- LDAP integration
- Single Sign On
- Mainframe discovery

Using BMC Atrium Discovery

The following topics introduce you to the tools and procedures for running discovery:

- Logging on to the system
- First steps in securing BMC Atrium Discovery
- Navigation
- Performing an initial discovery run
- Examining scan results
- Adding credentials
- Installing the Windows proxy manager and proxies
- Rescanning with credentials
- Scheduling a discovery
- Excluding ranges from discovery
Enabling other users
Providing your feedback

Logging on to the system

You must log on to BMC Atrium Discovery before you can use it to discover your IT infrastructure.

Before you begin

Before you start using BMC Atrium Discovery, your BMC Atrium Discovery Administrator will give you a user name and password and the URL of the BMC Atrium Discovery system at your location. Your user account and password may be the same as you use throughout your organization (single sign-on) or it may be specific to BMC Atrium Discovery.

The BMC Atrium Discovery appliance can be configured to use https rather than http. If you then specify a login URL which uses http, you are automatically redirected to the https page. See HTTPS Configuration for more information.

To log on to the system

1. Enter the BMC Atrium Discovery URL in your browser’s address bar. It might look like the following example:

   http://ip-address/

   Where ip-address is the address of the virtual machine, which is displayed when you log in to the virtual machine hosting the BMC Atrium Discovery using VMware console. If you received the logon URL from your administrator, it might have the host name instead of the IP address.

   The BMC Atrium Discovery logon screen is displayed.

2. Enter the username and password provided by your system administrator, or if this is the first time that anyone has accessed the appliance, use the following default values:
   - User name: system
   - Password: system
3. If you are using the Community Edition, agree to the terms and conditions of the Community Edition license by selecting the I accept the Terms and Conditions of use check box. To review the license, click the Terms and Conditions link.

4. Click Login.

5. If this is the first time that anyone has accessed the appliance, the Change Password for ADDM System User screen is displayed.

6. Change the system user password, to do this, complete the following fields:
   - Current password - enter the password you used to log on.
   - New password - enter your new password.
   - Verify password - enter your new password again to verify it.

7. Change the password for the command line user root. This can only be done if the default password has not been changed. A message is displayed if the password has been changed
   - New password - enter the new root password.
   - Verify password - enter the new root password again to verify it.

8. Change the password for the command line user tideway. This can only be done if the default password has not been changed. A message is displayed if the password has been changed
   - New password - enter the new tideway password.
   - Verify password - enter the new tideway password again to verify it.

9. It is strongly recommended that you secure the credential vault with a passphrase. If you choose not to, clear the Set Passphrase checkbox. Otherwise:
   - New password - enter the new credential vault passphrase.
   - Verify password - enter the new credential vault passphrase again to verify it.

10. Click Apply to make the changes.

The main BMC Atrium Discovery home page displays (see Viewing the home page).
Password quality

By default, the UI password that you enter must have at least one lowercase letter, one uppercase letter, one numeric character, and one special character. It must also contain a minimum of six characters. You can change this requirement by configuring password policies. See Managing security policies in the BMC Atrium Discovery Configuration Guide.

By default, the command line passwords that you enter must have at least one lowercase letter, one uppercase letter, one numeric character, and one special character. They must also contain a minimum of six characters, at least 4 characters must have been changed against the last password, and there must be no more than three repeated characters.

First steps in securing BMC Atrium Discovery

Once you have installed and logged into the BMC Atrium Discovery UI, you can make some initial configuration changes to improve the security of the system. If you have sufficient permissions to configure security settings, for example you are logged in as the system user, a Securing Your Appliance checklist is provided on the Home page default dashboard.
When you first log onto the UI, you are required to change the system user password and the passwords for the tideway and root command line users. These checklist items are shown as changed. The additional steps you should consider are enabling HTTPS access to the UI and redirecting HTTP access requests to HTTPS. Once you have configured these, the checklist updates (you may need to use CTRL+F5 to force refresh) and you are provided with a Dismiss button which you can click to remove the checklist from the dashboard.

The BMC Atrium Discovery appliance is designed and built with security in mind; a description of the security considerations and the types of measure taken are described in the Security section.

Navigation

This section provides an overview of logical modules of BMC Atrium Discovery that you can reach using the top navigation bar, and provides guidelines on the following:

- Using the Home page
- Applications
- Infrastructure
- Discovery
- Reports
- Administration

Using the Home page

The BMC Atrium Discovery home page displays predefined dashboards that provide a consolidated overview of certain aspects of your modeled IT infrastructure. Dashboards are particularly useful for highlighting standards and exceptions. For example, you can highlight under-used servers, or a breakdown of the operating systems on which Apache webservers are running.

When you initially log on to BMC Atrium Discovery, the default dashboard screen displays.
The default dashboard contains a Welcome to ADDM channel which describes the new features in the release. Should you eventually tire of reading about the new features, you can dismiss the channel by clicking **Dismiss**. This is a per-user setting, so dismissing the channel will not deprive other users of this happy reading experience. Unfortunately, you cannot restore your own Welcome to ADDM channel once it is dismissed.

On subsequent logins, the most recently used dashboard is displayed. On any user-defined dashboards, each section can be selected by clicking and holding the heading bar and can then be moved around the page. When you move the section to a new position, the outline is moved and the other sections rearranged to accommodate the new position.

You can also remove a section by clicking the **close** icon in the top right corner of the section. You can refresh the section with the refresh icon to the left of the **close** icon. The changes that you make are immediately available to all other users.

You can configure the dashboards to provide your own custom home page. The dashboards provided are described in Types of dashboards, and their use and how to customize them is described in Using and customizing dashboards.

The standard installed dashboards cannot be edited, though they can be copied, and the copy can be edited.

**Applications**

With **collaborative application mapping** you can understand the key relationships between how your business operates and the infrastructure that supports it and analyze how the hardware and software availability impacts your business services and applications. In the Application section of BMC Atrium Discovery you can:

- learn the basics of Application Mapping.
- model business services and applications and create patterns that automates the discovery of similar business services and application instances. See Knowledge management for more information on using the patterns.
- import existing application mapping definitions.
• review any of the application change impact reports that show how any network device, host, host model, OS and software impacts business applications they are connected with.

Infrastructure

In the **Infrastructure** section you can view a summary of the infrastructure items held in the system, the infrastructure items that are owned by you, as well as a number of infrastructure-related reports.

- item lists, where you can drill-down to view detailed information on any item (for example, the details of a host or a software instance and the relationships between objects),
- dashboard view, where you can drill-down to the item list view,
• visual viewer that displays automatically grouped objects in a convenient way.

**Infrastructure reports**

**Infrastructure explorer for viewing automatically grouped objects**

**Discovery**

From the discovery tab you can:

• manage and view the **Discovery status**
• set up **Credentials** to access host systems and obtain details of processes running
• understand how BMC Atrium Discovery determines Software Instances and Business Application Instances using additional patterns
• set up integration with external data sources (for example, databases or CMDBs),
• set up SNMP recognition rules
• view the Discovery reports
• download additional tools that help with discovery of Windows-based systems, secured or isolated networks.

Reports
The Reports menu in the top navigation bar opens the main reports page with a number of predefined queries that you can run as a list or dashboard reports. For more information about these and other available reports see Reporting basics.

Administration
The Administration page provides access to many commonly required setup operations. To access the Administration page, click Administration on the primary navigation bar. View-only tabs are also provided which enable you to view the current configuration of appliance identification, support information, software, hardware, network interfaces and routing. For more information about common administration operations see Navigating the administration interface.

Performing an initial discovery run
This section provides guidelines for an initial snapshot discovery without credentials. This produces a list of potential hosts that can be discovered.

Before you begin
Ensure you have the list of IP addresses in your network to scan, as well as IP addresses that should not be scanned and discovered for any reason.

To start discovery

1. Log on into BMC Atrium Discovery as the discovery user.
2. Click Discovery in the primary navigation bar.
   The Discovery status page is displayed.

4. Ensure that the **Snapshot** is selected.

5. Enter the IP address information in one of the following formats:
   - IPv4 address (for example 192.168.1.100). Labelled **v4**.
   - IPv4 range (for example 192.168.1.100–105, 192.168.1.100/24, or 192.168.1.*). Labelled **v4**.

   **Note**

   Scanning the following address types is not supported:
   - IPv6 link local addresses (prefix fe80::/64)
   - IPv6 multicast addresses (prefix ff00::/8)
   - IPv6 network prefix (for example fda8:7554:2721:a8b3::/64)
   - IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into *pills*, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning**: You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is
useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.
Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

---

**Information**

Pills are not currently supported in Opera.

6. Select **Sweep Scan** level for the discovery run. In sweep scan mode, the discovery is trying to determine what is at each endpoint in the scan range.
7. Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown.
8. For multitenant deployments, select the company name to assign to the discovery run. This drop-down list is only displayed if **multitenancy** has been set up. If (No company) is displayed, or a company name that you were expecting is missing, refresh the list by clicking **Lookup Companies** on the CMDB Sync. See **multitenancy** for more information about this feature.
9. Click **OK**.

The **Currently Processing Runs** tab is displayed with the new discovery run.

10. If discovery is running, the scan will start immediately.
    If the scans are stopped and you see the **Discovery is Stopped** message, you can enable discovery by clicking **START ALL SCANS**.

    The screen is automatically refreshed to show the status of the discovery process. When the scan is complete, you can find the discovery scan details and results on the **Recent Runs** tab.

Next steps

See **Examining scan results**.
Examining scan results

After a discovery run has completed, you can examine the results and look for potential hosts in the range that was scanned. You can access the scan results in several ways, such as from the Reports page.

1. To show the reports that are available on the discovered data, click **Discovery Reports**. The left side of the page shows the amount of information discovered in the credential-free scan, as shown in the following illustration.

2. To show the list of discovery runs, click the **Discovery Runs** icon.

3. To show the discovery run details, click on the line with the necessary discovery run information (reference).

4. To display the reports that are available on the Discovery run, click **Reports** and select **Possible Endpoint Host Devices (Detailed)** from the drop-down list.

5. Click the chart icon in the OS Class column heading.

6. From the menu, click **Pie Chart**. A chart displays the breakdown of the scan by OS class, as shown in the following illustration.

7. Continue based on the dominant OS class. This enables you to model a larger proportion of your environment in the first instance, and then return to the other OS class for more complete modeling.
   - If UNIX is the predominant OS class in the scanned environment, you can continue scanning by entering credentials to gather more information. Ideally, these credentials are valid for a range of computers and have sufficient rights to run system-level commands to discover richer data. Adding a credential is the same whether you add it for UNIX systems or Windows systems accessed through a Credential Windows proxy. You do not need to add credentials for systems accessed through the Active Directory Windows proxy because it uses the permissions of the user it was installed as. For more information, see Configuring host login credentials.
If Windows is the predominant OS class, you must download and install a BMC Atrium Discovery Windows proxy to perform Windows-specific discovery tasks. Go to Installing the Windows proxy manager and proxies.

Adding credentials

During discovery, the BMC Atrium Discovery system attempts to access host systems to obtain details of processes running. Credentials including IDs and passwords, and credential-like entities (Windows proxies and SNMP credentials) for different access methods, can be stored on the system to allow the required level of access. You can set up the following:

- **Login credentials** (user IDs and passwords) for interactive log-in to different host systems.
- **Windows proxies** used to discover Windows systems.
- **SNMP credentials** used on particular host systems.
  
  SNMP queries are only tried if an attempted login fails and if the SNMP port (UDP 161) is open on the target host.
- **vCenter credentials** used to discover VMware ESX and ESXi hosts by querying the vCenter server, including the management system vCenter credentials that are used to log into the vCenter. This view of the vCenter credentials is populated when patterns containing vCenter queries are activated.
- **vSphere credentials** used to discover VMware ESX and ESXi hosts, including the management system vSphere credentials that are used to log into the vSphere API. This view of the vSphere credentials is populated when patterns containing vSphere queries are activated.
- **WBEM credentials** used to discover storage systems managed using WBEM.
- **XML API credentials** used to discover devices with XML based APIs such as HP iLO devices.
- **Database credentials** used to log into databases and query their content.
- **Middleware credentials** used to query middleware such as web and application servers.
- **Mainframe credentials** used to discover mainframe computers.
- You can also run and view the progress of Device Credential Tests, as well as see the credentials that match the specific IP address or IP address range.

About credential storage

BMC Atrium Discovery stores all passwords in a secure credential vault. Secure credential vaults are generated using public and private key pair when the appliance is commissioned. The contents of the vault is encrypted, and can be secured using a passphrase.

User accounts on the target system

When creating a user account (the account that BMC Atrium Discovery logs into to discover a host) on a target host, you should ensure that the full path to the shell is entered in the user's profile. If you do not do this, the credentials will be considered invalid.

For example: SHELL=/bin/sh
Shell support

BMC Atrium Discovery is tested to work with Bourne and Bourne-compatible shells. Support for other shells such as the Korn shell is best effort only. The product has been sporadically tested and might work but with known issues and we might not fix bugs that affect these shells.

Installing the Windows proxy manager and proxies

To discover Windows hosts, BMC Atrium Discovery uses one or more *Windows proxies*. Once the BMC Atrium Discovery appliance decides that a discovery target is running a Windows operating system, it uses a proxy to interrogate the target. Often, the proxy is also responsible for providing authentication and authorization of the discovery activities.

There are two types of proxy:

- **Active Directory Proxy** — runs as an Active Directory user, and uses those user credentials to connect to Windows hosts within the Active Directory domain. Credentials are not stored in the BMC Atrium Discovery credential vault.
- **Credential Proxy** — runs as a local administrator user. Credentials are stored in the BMC Atrium Discovery credential vault and are provided to the proxy as required.

A single Windows host may run both types of proxy. To handle complex Active Directory environments, it is possible to run multiple Active Directory proxies as different users. The Active Directory proxy can also be used in a legacy Windows Workgroup environment to connect to workgroup members using the proxy's workgroup credentials.

The *Proxy manager* is used to manage the running proxies and their configuration, and to establish secure connections with approved BMC Atrium Discovery appliances.

Installing and running Windows proxies

### Installing or upgrading Windows proxies where anti-virus software is installed

Before installing Windows discovery proxies you should either disable the anti-virus software or configure it to exclude *RemQuery* from triggering a virus alert. You can enable the anti-virus software once the Windows proxy has been installed.

The Windows Proxy installer is downloaded from the BMC Atrium Discovery appliance user interface. Visit *Discovery > Tools* and download the installer. Both kinds of proxy, and the proxy manager, are in the same installer.
The installer prompts for the installation directory and whether to create start menu items. It also permits you to choose whether to create Active Directory and Credential proxies. In each case, credentials for a suitable user may be provided. If you do not create the proxies, or you do not enter credentials at this stage, you may do so later using the Proxy Manager. If proxies are created at this stage, the installer gives the option of registering the proxies with the appliance from which the installer was downloaded. Registration in this way opens a web browser displaying the appliance UI, and is therefore only possible if the Windows proxy host has web access to the appliance.

To modify the proxies that are running, or add new ones, run the Proxy Manager from the Start menu, by default under **BMC Software > ADDM Proxy > Proxy Manager**.

The Active Directory proxy must be given the credentials of an Active Directory user. Ideally the user should be a domain administrator. If that is not possible, the user must have a range of permissions, and discovery capabilities will be limited. The Credential proxy should run as a local Windows user with administrator privileges.

Each proxy listens on a particular port. The default ports are 4321 for the Active Directory proxy and 4323 for the Credential proxy. Whichever ports you choose must be reachable from the BMC Atrium Discovery appliance, so the Windows firewall and any intermediate network firewalls must allow connections from the appliance’s IP address.

When scanning a Windows discovery target, the BMC Atrium Discovery appliance chooses a Proxy Pool based on the target’s IP address. Often, a pool contains a single proxy, but if the proxy becomes a bottleneck, a pool can contain several proxies.

**Connecting a Windows proxy to an appliance**

For security, the proxy and the appliance must exchange certificates and each end must confirm the connection.

If it is possible to access the appliance web user interface from the Windows machine running the proxy, you may set up the connection between appliance and proxy from the Proxy Manager user interface:

1. In the Proxy Manager, choose **Edit menu > Known Appliances**
2. Press the green plus icon to add an appliance
3. Enter the appliance address, and select **Contact**
4. The proxy manager exchanges certificates with the appliance and displays the appliance fingerprint. Assuming the fingerprint is correct, select **Register**
5. A browser window opens, displaying the Create Proxy page. Confirm the details and select **Apply**

If there is no web access from the Windows proxy machine to the appliance, the proxy connection must be initiated from the appliance user interface and confirmed in the Proxy Manager:

1. In the appliance web user interface, visit **Discovery > Credentials > Windows Proxies**
2. If need be, add a new Proxy Pool using the **Add...** button
3. In the chosen Proxy Pool, select **Actions > Add Windows Proxy**
4. Enter a proxy name and address, and select **Apply**. The user interface shows **Approve this appliance in the proxy manager**
5. In the Proxy Manager, choose **Edit menu > Known Appliances**
6. The new appliance is shown in orange with a status **Pending approval**
7. Double click the entry then choose **Approve**

**Rescanning with credentials**

After you have added credentials, you can run the first scan again.

1. Click the **Discovery Status** button to go to the Discovery page.
2. Select the **Recent Runs** tab.
3. For the first test scan, click **Rescan Now**.
   - The same scan runs again, with login credentials for the IP addresses. You can now view the results of the scan and look for details of hosts in the range that you have scanned.
4. Click **Discovery Reports** to display the reports that are available on the discovered data.
   - Comparing the results to the Discovery reports page after the **credential free scan**, you can see that significantly more information was obtained on the later scan.
   - Now you can examine the hosts that BMC Atrium Discovery has discovered.
5. Click the **Infrastructure** tab.
6. Click the Hosts icon to display a list of discovered hosts.
7. Click on a host to view it.
   - You have now successfully scanned the network segment with and without credentials and have viewed a discovered host.

**Destroying hosts**

If you have reached the maximum number of devices (60) permitted in the Community Edition of BMC Atrium Discovery, discovery stops and you cannot start any new discovery runs. Ideally, you should only target discovery runs at the devices that you are interested in are scanned.

If you need to carry out further work and scan additional hosts, you must destroy some existing ones so that you can start discovery again.

1. Click the **Infrastructure** tab.
2. Click the Hosts icon to display a list of discovered hosts.
3. Select the hosts that you want to destroy by clicking the selection box at the left of the host row. You can select multiple hosts. You can also select all hosts by clicking the selection box at the left of the heading row.
4. Click **Destroy**.
5. Click **OK** to destroy the hosts, or click **Cancel** to return to the hosts page without destroying any hosts.
Scheduling a discovery

1. On the discovery page, click **Add New Run**. For Consolidation appliances, click **Add New Local Run**.
2. Select **Scheduled**. The dialog displays Frequency, Start, and End menus.
3. Enter the information for the scheduled scan in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| **Range**  | Enter IP address information in one of the following formats:  
• IPv4 address (for example 192.168.1.100), Labelled v4.  
• IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*), Labelled v4. |

**Note**

- Scanning the following address types is not supported:
  - IPv6 link local addresses (prefix fe80::/64)
  - IPv6 multicast addresses (prefix ff00::/8)
  - IPv6 network prefix (for example fdad:7554:2721:a8b3::/64)
  - IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into *pills*, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the *Showing n of n* label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.
- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| **Level**  | Select the level for the discovery run. This is either:  
• **Sweep Scan** — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type.  
• **Full Discovery** — Retrieves all the default info for hosts, and complete full inference. |
| **Label**  | Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown. |
| **Company** | Select the company name to assign to the discovery run. This drop-down list is only displayed if multitenancy has been set up. If (No company) is displayed, or a company name that you were expecting is missing, refresh the list by clicking **Lookup Companies** on the **CMDB Sync**. See multitenancy for more information about this feature. |
| **Frequency** | |
### Field name | Details
--- | ---
Select a frequency for the discovery run to be performed. This can be:  
• Weekly by days of week  
• Once per week  
• Monthly by day of month  
• Monthly by week of month.

### Start
Based on the selected scan frequency, you are presented with different options to start scheduled scans. Discovery must be running at this time.

- **Weekly by days of week**: You are provided with buttons for each day and drop down menus for the start time in hours and minutes. You can select one or more days from the day buttons. The selected buttons appear with a Yellow border.
- **Once per week**: You are provided with drop down menus to select the day of the week and the start time in hours and minutes.
- **Monthly by day of month**: You are provided with drop down menus to select the day of the month and the start time in hours and minutes.
- **Monthly by week of month**: You are provided with drop down menus to the select the week, the day of the week, and the start time in hours and minutes.

For example, to start a scheduled weekly discovery run which starts on Friday at 19:30 hrs and continues until Saturday, you will do the following:

1. Select the scan frequency as **Weekly by days of week**.
2. Click on the F and S day buttons.
3. From the time drop down menu, select 19 hours and 30 minutes.

### End
You can choose end a scheduled scan when it is completed by selecting **when completed**. Alternatively, use the available option to end the scan which is based on the selected scan frequency. If the duration of the end time expires before the scan has completed, then the scan is suspended until the next scheduled time that the scan occurs. The scan resumes from the point where it was previously suspended.

- **Weekly by days of week**: You are provided with the drop down menus to select the end time in hours and minutes.
- **Once per week**: You are provided with the drop down menus to select the day of the week and time to end the scan.
- **Monthly by day of month**: You are provided with the menus to select the day of the month and time to end the scan.
- **Monthly by week of month**: You are provided with the drop down menus to the select the number of days and the time within which the scan must end.

For example, to end a scheduled weekly discovery run which starts on Friday at 19:30 hrs and continues until 21 hours 30 minutes on Saturday (see the example for the Start field above), you will select 21 hours and 30 minutes from the time drop down menu.

4. Click **OK**. The **Scheduled Runs** tab is displayed with the new scheduled discovery run.

To add another scan to the page, click **Add New Run**.
To delete any existing scheduled scans, select the entry and click **Delete**.

**Next steps**
For more information, see **Scanning IP addresses or ranges**.
Excluding ranges from discovery

There might be some IP devices in your network that you either permanently do not want to be scanned, or want to exclude those from a scan at a scheduled time. For example, you might have some legacy applications which will only run on old hardware which might be regarded as _fragile_.

You can add this device as a single IP exclude, or as an Exclude Range.

- **To add an Exclude Range**
- **To edit an existing exclude range**
- **To enable or disable an exclude range**

**To add an Exclude Range**

You can add the following types of exclude ranges:

- Permanent
- Scheduled

**To add a permanent exclude range**

1. On the Discovery Status page, click **Add New Exclude**.
2. Enter the information for the exclude range in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  

  - IPv4 address (for example 192.168.1.100). Labelled v4.
  - IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4.

As you enter text, the UI divides it into _pills_, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the _Showing n of n_ label below the _Range_ field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the _Range_ field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

_Pills are not currently supported in Opera._

<table>
<thead>
<tr>
<th>Label</th>
<th>Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>A free text description of the exclude range.</td>
</tr>
</tbody>
</table>
3. Click **OK**.
   The **Excluded Ranges** tab is displayed with the new exclusion.

**To add a scheduled exclude range**

1. On the Discovery Status page, click **Add New Exclude**.
2. Enter the information for the exclude range in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  
  • IPv4 address (for example 192.168.1.100). **Labelled v4**.  
  • IPv6 address (for example 2001:500:1187:203:baff:fe44:91a0). **Labelled v6**.  
  • IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). **Labelled v4**. |

**Large scheduled exclude ranges**

If you specify a scheduled exclude range which blocks 65535 IP addresses or more of a scan range, new IP addresses from that scan are not added to the scan queue until at least one of the blocked IP addresses is allowed to scan.

As you enter text, the UI divides it into **pills**, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

- **Pills are not currently supported in Opera.**

<table>
<thead>
<tr>
<th>Field</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the exclude range.</td>
</tr>
<tr>
<td>Frequency</td>
<td>Add text</td>
</tr>
</tbody>
</table>
| Start | Based on the selected frequency of the scheduled exclude, you are presented with the following start options:  
  • **Weekly by days of week:** You are provided with buttons for each day and drop down menus for the start time in hours and minutes. You can select one or more days from the day buttons.  
  • **Once per week:** You are provided with drop down menus to select the day of the week and the start time in hours and minutes.  
  • **Monthly by day of month:** You are provided with drop down menus to select the day of the month and the start time in hours and minutes.  
  • **Monthly by week of month:** You are provided with drop down menus to select the week, the day of the week, and the start time in hours and minutes. |
For example, to start a scheduled weekly exclude which starts on Friday at 19:30 hrs and continues until Saturday, you will do the following:
  a. Select the exclude frequency as **Weekly by days of week**.
  b. Click on the F and S day buttons.
  c. From the time drop down menu, select **19 hours and 30 minutes**.

Based on the selected frequency of the scheduled exclude, you are presented with the following end options:

- **Weekly by days of week**: You are provided with the drop down menus to select the end time in hours and minutes.
- **Once per week**: You are provided with the drop down menus to select the day of the week and time to end a scheduled exclude.
- **Monthly by day of month**: You are provided with the menus to select the day of the month and time to end a scheduled exclude.
- **Monthly by week of month**: You are provided with the menus to select the number of days and the time within which a scheduled exclude must end.

For example, to end a scheduled weekly exclude which starts on Friday at 19:30 hrs and continues until 21 hours 30 minutes on Saturday (see the example for the Start field above), you will select **21 hours and 30 minutes** from the time drop down menu.

**To edit an existing exclude range**

You can edit an existing exclude range. If the exclude range is currently in progress, it is automatically cancelled when you edit it.

1. From the **Exclude Ranges** tab of the Discovery Status page, click the exclude range that you want to edit.
   All of the fields are editable.
2. Make the required changes and click **OK**.

**To enable or disable an exclude range**

You can enable or disable an existing exclude range, whether it is permanent or scheduled.

From the **Exclude Ranges** tab of the Discovery Status page, select individual, multiple, or all exclude ranges to enable or disable. If the exclude range is currently in progress, it is cancelled if you make and apply any changes.

**Enabling other users**

Typically, when you want to enable additional users to access BMC Atrium Discovery, you should log in as a user with the appropriate privileges. In addition to the system user, the following additional users are configured in BMC Atrium Discovery, but are not enabled by default:

- **admin**
- **appmodel**
- **discovery**
Use the *system* user only for configuration tasks which require system privileges.

**To enable other users**

1. Click the **Administration** tab.
2. In the Security Policy section of the Administration page, ensure the "Disabled Accounts can be reactivated" option is set to Yes.
3. In the Security section of the Administration page, click the **Users** icon.
4. For each user, click the **Set Password** link.
5. On the Set Password page, enter the new password in each text entry field and click **Apply**.
6. After you have changed the passwords for each user, log off from the system user account by clicking the **logout** icon at the top right of the page.

**Providing your feedback**

BMC Atrium Discovery is designed with your needs in mind and your feedback is important. When you find anything in BMC Atrium Discovery that you particularly like, or when anything does not work for you or could better meet your needs, you can send us your feedback directly through the UI.

**To send your feedback**

1. Hover over the blue ribbon on the bottom right of any UI page and click **Feedback**. The User Feedback page is displayed. The **Current page** label supplies context for your feedback.

![User Feedback](image)

2. Select one of the following available choices from the **What I think** drop down:
   - I like it!
   - I don't like it!
   - This could be different.
   - This does not work for me.
3. Provide your feedback in the text entry field.
4. Click **Submit** to send your feedback to BMC.

**Next steps**

The following topics contain information on more advanced configuration options:
• To avoid exposing credentials in the information about discovered processes (for example, the command used to launch the process), learn how to mask sensitive data.
• To examine or modify discovery commands, follow the guidelines provided in Managing the discovery platform scripts.
• Some commands return more information when they are launched with the superuser credentials. You can authorize log on with the privileged credentials for such commands by adding privileged execution to the related discovery script.
• To enable custom port scanning for OS fingerprinting, follow the guidelines here.
• To simulate a discovery of a system that is unreachable for the BMC Atrium Discovery machine, you can use manual scanning using the discovery scripts. For Windows systems, you can use the Standalone Windows scanning tool.
• To improve the discovery you can monitor the success rate of the credentials, troubleshoot access issues, and see discovery conditions for information on potential access issues.
• To enrich the discovered data with the information available in external databases, see Integration points.
• To enable centralized management and decentralized discovery in firewalled environments and restricted networks, use Consolidation.
• To allow BMC Atrium Discovery connect with the required level of access to the discovered systems in your network, see Credentials. You might set up various stored credentials, including IDs and passwords, and credential-like entities (Windows proxies and SNMP credentials) for different access methods.
• To learn about the ports and protocols BMC Atrium Discovery uses to scan your IT environment, see Discovery communications.
Security

This document provides a brief overview of the security aspects of BMC Atrium Discovery. It is intended to provide network administrators with the information required to run BMC Atrium Discovery in their environment. It also provides the information required to enable security teams to verify that BMC Atrium Discovery is secure and does not compromise the security of their network.

- Security in BMC Atrium Discovery
- An appliance-based solution
- Appliance hardening
- Understanding security audits
- Information security
- System communications
- Discovery communications
- Firewall Port Summary
- DISA Secure Technical Implementation Guidelines
- Running in FIPS compliant mode
- PCI Data Security Standard compliance

Security in BMC Atrium Discovery

BMC Atrium Discovery is an appliance-based tool which automates discovery of business applications, maps them onto the underlying physical and virtual IT infrastructure, and determines the critical dependencies between them. BMC Atrium Discovery's model-driven, data center indexing techniques cut across previously disparate silos of configuration information, automatically populating and maintaining a data store of the discovered state of Configuration Items (CI) and dependency information.

BMC Atrium Discovery automates many system administrator and application management team tasks and also stores customer-sensitive data. At its core, BMC Atrium Discovery ensures the confidentiality and integrity of the discovery processes as well as the indexed data itself.

This document is intended to provide network administrators with the information required to get BMC Atrium Discovery working in their environment. It also provides the information required to enable security teams to verify that BMC Atrium Discovery is secure and does not compromise the security of their network.

Automated discovery in secure environments

BMC Atrium Discovery offers a powerful solution to index the infrastructure and map the business services of the large and complex environments typical of Fortune 1000 enterprises.
Several techniques could be used to gather data, such as port scanning, protocol probing, agent-based monitoring and remote login. However, credentials are required to achieve the accurate, trusted, and detailed configuration data discovery that enterprises require to manage IT.

Why? Simply put, because this is what information security is about: protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction. Enterprises spend millions to secure their systems and ensure their access is protected by using credentials to authenticate users. Providing this type of permission ensures that BMC Atrium Discovery does only what it is authorized to do.

However, there is a side effect to this need for granular visibility. IT operations are organized in functional groups, or silos. Applications span silos, so there is rarely one team in charge of all credentials. In fact, too frequently they are not sure whether they can find all the credentials. IT administrators expect that automated discovery tools will solve problems caused by the lack of knowledge about where their systems or applications are (including their credentials). Additionally, in large and complex environments some parts of the networks are segregated for confidentiality, business or even historical reasons such as acquisitions, creating even more silos.

BMC Atrium Discovery offers a unique approach to these challenges and concerns by providing:

- A robust and secure delivery platform
- Clear deployment requirements

An appliance-based solution

A frequently asked question in the technology industry is whether one should favor appliance based solutions (hardware or virtual) or software-based solutions (which need to be installed and configured); a valid question as products in the same category often take these two different approaches. While the cost, performance, maintenance and support for these approaches are similar, the differences in security are often a source of concern.

Appliance advantages

Anything running on a host can be considered a potential security risk. If a component is not actually required then it is safer not to install it. Vulnerabilities in the many tools and utilities installed and running in a default installation of an OS are known and exploited. The appliance approach provides a tightly controlled system in which only the essential tools and utilities are installed. These tools and utilities, including the Red Hat Enterprise Linux (RHEL) 6 OS, are hardened to allow only authorized access and ensure the integrity of the system. See Appliance hardening for more information.

Security updates

A considerable advantage of the appliance approach over a software solution is a known and understood system in which the interaction between components is designed and knowingly limited to that design. When patches to the RHEL OS are released, BMC Software check to see whether
they are appropriate to the appliance. Many are inappropriate due to the subset of packages used in the appliance. Where a patch is appropriate it is tested and rolled into the next available OS upgrade, or product release; urgent updates are released as a Hot Fix.

BMC provides regular upgrades to the BMC Atrium Discovery OS each month; each upgraded package is checked to see whether they are appropriate to the appliance. See OS upgrades for more information.

---

**Do not download and apply Red Hat OS patches**

It is most important that OS patches released by Red Hat are not downloaded and applied to the appliance; this might result in reduced rather than enhanced security. For example, a patch might reinitialize a service, modify security configurations, or change kernel parameters, all of which can cause unexpected behavior.

---

**Patch versioning**

Red Hat does not increment the base version of any of the packages until the whole release is incremented. Instead they continuously apply security patches. This means simpler security scanners can report false positives as they only look at the base version of the packages.

**Software-based solutions**

In contrast, software-based solutions are generally installed on servers that are supplied by customers. This approach has advantages as it provides the customer full control over how to implement, configure and support the solution. However it includes several aspects to consider which can impact the security of the system. Vendors often specify a minimum set of OS packages that are required to support the software-based solutions, placing customers in the difficult position of choosing what is needed versus what is not. Not only does this allow potential security vulnerabilities, it also makes the task of hardening the system far more complicated.

Finally, the more packages there are on servers, the more security patches a company must monitor to secure these servers. Since the customer generally provides the server, the burden of monitoring security patches falls on the customers.

**Appliance hardening**

The following measures are taken to harden the BMC Atrium Discovery appliance when it is built:

- Build the OS using only a small number of packages, all of which are required
- Only the required services are enabled
- Firewall specifically tuned for the appliance
- Unnecessary user accounts are removed
- Disable telnet and ftp (access is through ssh only)
- No remote logins as root
• Set specific kernel parameters such as ICMP echo broadcast
• Set permissions on logging, cron, and configuration to require a privileged user
• Mount options configured to permit only certain operations on specific partitions
• Password quality criteria set
• Remove SETUID privileges from certain applications

The appliance is equipped with its own baseline monitoring system (based on the open source Tripwire product) which can be configured to automatically take action in case of unauthorized changes, such as shutting down the appliance or disabling access.

The complete package list is included in the Release Notes rather than this document as they can be upgraded between minor releases.

User management

BMC Atrium Discovery application’s internal user management service offers all the features required to support ISO 27002 guidelines, specifically:

• Account management
• Password management policies (strength, reuse, lifecycle)
• Granular groups permissions
• Account blocking after authentication failures
• Automatic account lockout (for example, an account not used for 60 consecutive days)
• Automatic session lockout (for example, a session left idle for more than 30 minutes)

Many firms have invested in identity and access management solutions to centralize user management and the permissions to the applications they can access. BMC Atrium Discovery can also integrate with a corporate LDAP solution such as Active Directory so that user accounts and group permissions can be managed directly from the LDAP. LDAP groups can be mapped as desired to BMC Atrium Discovery groups to simplify overall administration.

BS7799, ISO 17799, and ISO 27002

BS7799 was an early two part information security standard.

• BS7799-1 was a code of practice which when adopted by ISO became ISO 17799. It was renamed ISO 27002 in 2007.
• BS7799-2 was the specification against which a system could be certified. This became ISO 27001.
Appliance firewall

The appliance firewall is pre-configured to ensure only the following incoming traffic is allowed. Windows proxy communication is always initiated from the appliance so is not listed here.

The open ports listed below are incoming TCP ports to the appliance.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Description</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>Secure Shell Login</td>
<td>For remote management of the appliance OS.</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>For accessing the appliance web user interface, if enabled.</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>For accessing the appliance secure user interface, if enabled.</td>
</tr>
<tr>
<td>25030-25032</td>
<td>CORBA over TLS</td>
<td>To enable appliance clustering.</td>
</tr>
<tr>
<td>25032</td>
<td>CORBA over TLS</td>
<td>To enable discovery consolidation.</td>
</tr>
</tbody>
</table>

The appliance approach provides a known and understood system in which the interaction between components is designed; the firewall is one of those components. Consequently the appliance is expected to have full control over the firewall. Local Linux system administrators should not make any changes to the appliance firewall as this can compromise the appliance security and any changes will be lost when the it is upgraded.

The only supported change to the appliance firewall is that required to install BMC PATROL. Where such changes are made, the default firewall is used as a fallback.

Where further monitoring or protection is required then it should be placed behind an additional firewall.

Windows proxy hardening

Windows discovery requires a Windows proxy or proxy running on a Windows host to provide the methods (WMI and RemQuery) of accessing Windows systems. The Windows proxy host should be configured to allow the following incoming traffic on the chosen ports.

The ports can be chosen in the proxy manager. The defaults are:

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4321</td>
<td>Used to connect to a Active Directory Windows proxy from the BMC Atrium Discovery appliance.</td>
</tr>
<tr>
<td>4323</td>
<td>Used to connect to a Credential Windows proxy from BMC Atrium Discovery appliance.</td>
</tr>
</tbody>
</table>

Penetration testing

To ensure BMC Atrium Discovery data integrity and confidentiality, the BMC Quality Assurance group performs a thorough assessment on each major and minor release.

UI penetration tests are made with IBM® AppScan®.
System penetration tests are made with Tenable Nessus. Bastille Linux is used in assessment mode to confirm the security configuration of the system.

**Limited/hardened Red Hat Enterprise Linux distribution and security scanners**

It is important to note that BMC Atrium Discovery does not include a full Red Hat Enterprise Linux build with all of its various packages. In order to improve the security of the product, BMC Atrium Discovery only includes those components needed for the operation of the product, rather than those required for a general purpose OS. Omitting unnecessary components decreases risk and increases the overall security of the product.

However, the fact that BMC Atrium Discovery doesn't include the full OS can often confuse general purpose security scanners. When the scanner checks the OS, it will report that it is missing patches for components that were never included in the distribution. For example, if BMC Atrium Discovery does not include component xyz, it certainly would not include a patch for that component. Since general purpose tools do not first check to see if the component for a patch is present, it simply reports the patch missing without realizing it would make no sense for it to be included on that server.

**Known false positives flagged by security scanners**

The following security issues have been flagged in the past by some security scanners. In each case they can be shown as not being applicable to BMC Atrium Discovery.

- Cyrus SASL Library Base64 Encoder Buffer Overflow — Cyrus IMAP is not part of the BMC Atrium Discovery appliance.
- LibPNG could cause denial of service — as there is no UI method of uploading PNG files, the exploit requires command line access as the tideway user.
- LibXML issues could cause crashes — as there is no UI exposure of the XML system, the exploit requires command line access as the tideway user.
- WLAN issue with Kernel — the exploit requires WLAN to be enabled and WLAN kernel extensions to be installed. Neither of these are installed on the appliance.
- OpenSSH X11 Port forwarding hijack — X11 is not installed on the appliance.
- OpenSSL Record of death — not applicable to the version of OpenSSL installed on the appliance.
- Sudo RunAs Group — not applicable to the version of sudo installed on the appliance.
- SQL injection errors — the data store does not use SQL.

The next section describes ways in which you can identify similar false positives.

**Understanding security audits**

Many customers employ a combination of security analysts (penetration and application testers) and software to find security vulnerabilities in the BMC Atrium Discovery virtual appliance. The virtual appliance consists of Red Hat Enterprise Linux (RHEL) and the BMC Atrium Discovery application software.
The goal of this page is to help you to understand the results your security audits, to determine which items are valid, and which are false positives, that is those which are incorrectly reported as vulnerabilities.

Operating System Audit

BMC Atrium Discovery customers typically run a 3rd party application to determine if there are any vulnerabilities in the OS. These tools, such as Qualys, eEye Digital Security Retina, and Tenable Nessus, check the versions of the packages installed against a known database of vulnerabilities. This database is a combination of their companies' intellectual property and those in the public vulnerability databases such as the Common Vulnerability Engine (CVE) - maintained by the MITRE corporation) and the National Vulnerability Database (NVD).

Often these tools are not aware of the method Red Hat uses to apply functionality and security fixes. As a standard, Red Hat chooses a fixed point to release a new major version, for example, RHEL 5.0. All packages included in the release are then fixed at the level of functionality available upstream at the time and Red Hat will only apply fixes to the versions. Upstream is the term used to refer to the open source projects that do the development work on the packages employed in Red Hat, such as the kernel.

For example, the ssh package for RHEL 5 is at version 4.3p2-36.el5. Red Hat have applied a number of fixes to this and the version at RHEL 5.7 is 4.3p2-72.el5.7.5.el5. Notice that the functionality version, 4.3p2, has not changed. The ssh version for RHEL 6.0 is 5.3p1-18.el6 as Red Hat have chosen 5.3p1 to be the base for RHEL 6.

This causes issues with some of the vulnerability assessment tools as they are not "Red Hat aware". Take for example the bzip2 package. The version installed at RHEL 5.7 is 1.0.3–6.el5.5. A vulnerability (CVE, NVD) was found in all versions of bzip2 before 1.0.6 that could cause a Denial of Service (DOS) attack. The tool would identify the fact that the Red Hat installation has version 1.0.3 installed, which appears to have the vulnerability. However, the fix was applied by Red Hat in version 1.0.3-6.el5.5. This is a false positive.

How to determine a false positive in an OS audit

The following examples describe the methodology you should employ in determining whether an alert is a false positive or not.

Example 1

This is an excerpt from the output of a vulnerability reporting tool:

<table>
<thead>
<tr>
<th>Risk Level</th>
<th>Sev Code</th>
<th>Category</th>
<th>CVE</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>Cat I</td>
<td>Local UNIX Security Audits</td>
<td>CVE-2009-0688</td>
<td>The CMU Cyrus SASL (Simple Authentication and Security Layer) library contains a buffer overflow vulnerability when encoding strings into BASE64 format. Successful exploitation could allow attackers to execute arbitrary code or cause the application using the Cyrus SASL library to</td>
</tr>
<tr>
<td>Risk Level</td>
<td>Sev Code</td>
<td>Category</td>
<td>CVE</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>----------</td>
<td>-----</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>crash.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td><em>Note:</em> This audit is for versions of Cyrus SASL obtained from asg.web. cmu.edu/sasl/ and might report false findings with vendor-specific backports of Cyrus SASL.</td>
</tr>
</tbody>
</table>

The first step is to check the details on the CVE page. To find this page, check the MITRE CVE site and search for the CVE number. Entering the CVE number (CVE-2009-0688) in a search engine should also find the CVE. The CVE shows that Red Hat have released a statement, a Red Hat Security Advisory (RHSA) describing the vulnerability:

**CVE-ID**

<table>
<thead>
<tr>
<th>CVE-2009-0688 (under review)</th>
<th>Learn more at National Vulnerability Database (NVD)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings</td>
</tr>
</tbody>
</table>

**Description**

Multiple buffer overflows in the CMU Cyrus SASL library before 2.1.23 might allow remote attackers to execute arbitrary code or cause a denial of service (application crash) via strings that are used as input to the `sasl_encoded64` function in lib/saslutil.c.

**References**

*Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.*

- CONFIRM: http://ftp.cs.cmu.edu/pub/cyrus-mail/cyrus-sasl-2.1.23.tar.gz
- CONFIRM: http://support.avaya.com/saplicate/security/A004-1914.htm
- CONFIRM: http://support.apple.com/kb/HT4577
- APPLE:APPLE-SA-2010-03-29-1
- DEBIAN: DSA-1807
- GENTOO: GLSA-200907-09
- URL: http://security.gentoo.org/glsa/glsa-200907-09.xml
- MANDRIVA: MDVSA-2009:113
- REDHAT: RHSA-2009:1116
- SUSE: SUSE:ISS-2009-124-01

The Red Hat Security Advisory is where Red Hat either release update information, or the Red Hat Bugzilla issue number. In either case they are generally preceded by REDHAT. Either click the link provided by CVE, and search for the RHSA number (RHSA-2009-1116), or search the Red Hat Bugzilla database for the issue number.

The RHSA tells you what version of the package the bug is fixed in.
The edition of Red Hat Linux we are running is referred to as Red Hat Enterprise Linux Server, all other editions can be ignored (Workstation, AS, ES, EUS, Long Life, WS, and so forth). Search for the x86_64 version (Intel/AMD 64-bit). See the example below:

Important: cyrus-imapd security update

Advisory: RHSA-2009:1116-1
Type: Security Advisory
Severity: Important
Issued on: 2009-06-18
Last updated on: 2009-06-18

Affected Products:
- Red Hat Desktop Workstation (x 5 client)
- Red Hat Desktop (x 4)
- Red Hat Enterprise Linux 4 (x 5 server)
- Red Hat Enterprise Linux AS (x 4)
- Red Hat Enterprise Linux ES (x 4)
- Red Hat Enterprise Linux ES (x 4.8 z)
- Red Hat Enterprise Linux EUS (x 5.3 x server)
- Red Hat Enterprise Linux Long Life (x 5.3 server)
- Red Hat Enterprise Linux WS (x 4)

CWEs (cve.mitre.org): CVE-2009-0688

Details

Updated cyrus-imapd packages that fix a security issue are now available for Red Hat Enterprise Linux 4 and 5.

This update has been rated as having important security impact by the Red Hat Security Response Team.
The details or descriptions provided in the RHSA and CVE contain much useful information. Often Red Hat will provide reasons why the specific package is affected. Occasionally the CVE will indicate that the issue is with a specific package, as shown in this example where the CVE states that the Cyrus SASL library that has the issue. Red Hat, however, have fixed the issue in the cyrus-imapd package rather than the cyrus-sasl package, the detail of which is described in the RHSA details section.

Now that the version of the package in which Red Hat have fixed the vulnerability is known, we must confirm that the package is updated in the latest version of BMC Atrium Discovery. In the following example, we can check whether or not the packages are installed. As all the affected packages listed on the RHSA page start with cyrus-imapd it is simple to search the entire RPM database and grep for the prefix rather than search individually:

```
tideway@disco01 ~>]
$ tideway@disco01 ~>] rpm -qa | grep cyrus-imapd
tideway@disco01 ~>]
```

The package does not appear to be installed. Performing a wider search just for cyrus shows the cyrus packages installed:

```
tideway@disco01 ~>]
$ tideway@disco01 ~>] rpm -qa | grep cyrus
  cyrus-sasl-lib-2.1.22-5.el5_4.3
cyrus-sasl-lib-2.1.22-5.el5_4.3
cyrus-sasl-devel-2.1.22-5.el5_4.3
cyrus-sasl-plain-2.1.22-5.el5_4.3
cyrus-sasl-2.1.22-5.el5_4.3
cyrus-sasl-plain-2.1.22-5.el5_4.3
tideway@disco01 ~>]
```
The cyrus-imapd package is definitely not installed, so the issue cannot affect BMC Atrium Discovery. This issue can be reported as a false positive.

**Example 2**

Sometimes it is not immediately obvious whether or not the issue affects Red Hat and careful attention needs to be paid to the CVE detail.

This is an excerpt from the output of a vulnerability reporting tool:

<table>
<thead>
<tr>
<th>Risk Level</th>
<th>Sev Code</th>
<th>Category</th>
<th>CVE</th>
<th>Description</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>Cat II</td>
<td>Web Servers</td>
<td>CVE-2010-0740</td>
<td>OpenSSL contains a vulnerability (i.e. &quot;record of death&quot;) when handling malformed records during TLS connections. Successful exploitation could allow attackers to cause denial of service conditions (i.e. daemon crash).</td>
<td>Upgrade software that is using vulnerable versions of OpenSSL libraries and/or upgrade OpenSSL to version 0.9.8n or newer.</td>
</tr>
</tbody>
</table>

*Note:* This audit is for versions of OpenSSL obtained from OpenSSL.org and might report false findings with vendor specific backports.

The first step is to check the details on the CVE page. Immediately it is evident that there is no REDHAT entry listed. The closest thing is Fedora which is another Linux distribution sponsored by Red Hat.
The report states that we should upgrade to version 0.9.8n or later. The following example shows checking the version currently installed.

tideway@disco01 ~]$ tideway@disco01 ~]$ rpm -q openssl
openssl-0.9.8e-20.el5 openssl-0.9.8e-20.el5

tideway@disco01 ~]$

In this example, the same version of the package is listed twice. This occurs when both 32 and 64 bit packages are installed.

The version running is 0.9.8e, which is definitely earlier than 0.9.8n, the version the CVE recommends. Red Hat however have not provided a fix. Possibly one was never required. Careful reading of the CVE description offers a clue.
The `ssl3_get_record` function in `ssl/s3_pkt.c` in OpenSSL 0.9.8f through 0.9.8m allows remote attackers to cause a denial of service (crash) via a malformed record in a TLS connection that triggers a NULL pointer dereference, related to the minor version number.

According to the CVE description the vulnerability exists in versions 0.9.8f through 0.9.8m. The virtual appliance is running version 0.9.8e so the vulnerability that was created in the changes from e to / does not affect our version. Again, the issue cannot affect BMC Atrium Discovery and can be reported as a false positive.

**Information security**

**UI security**

The passwords used to access the BMC Atrium Discovery UI (such as for the system user) are salted, hashed with SHA-256 and stored in a file.

**Credential vault security**

The credentials used to log in to discovery targets, synchronize to the CMDB, and export data using adapters are stored in a vault that is encrypted with a default passphrase when the appliance is built. The vault provides a secure mechanism for storing credential information. Only users with Discovery or Administration privileges have read/write access to the vault, with read access limited to non-sensitive information only (passwords can never be seen in the UI or at the command line). The content of the vault is secured using 256 bit AES encryption in CBC mode.

The default vault passphrase is persisted on the appliance, and is common to all appliances, therefore it is highly recommended, and considered security best practice, to secure the vault with a manually entered passphrase. When the passphrase is set, the vault is automatically in a locked state when the appliance starts, and requires the passphrase to be unlocked. The encryption key used for encrypting the vault is derived from the passphrase. The passphrase is not stored anywhere on the appliance, and must be supplied by a user with sufficient privilege. If the passphrase is lost, the contents of the vault cannot be recovered. Without a manually entered passphrase the vault is only guarded against casual inspection, in which case vault security is dependent on Linux command line security.

The default passphrase used is a random string of 64 characters/512 bits to generate a 256 bit key. If you decide to use a manually entered passphrase you should ensure that it is of at least a similar complexity, or that it is changed at regular intervals.

A “Security Best Practice” may be to defer credential management to the in house security team who would manage credentials according to their own requirements. Permission could be granted for the security team to update the passwords stored in the vault, and for other users to run discovery using the stored passwords.
Sensitive data filters
Data returned from discovery targets can contain sensitive data. For example, the command used to start the process might contain a clear text password. This data is stored in a DiscoveredProcess node and could be viewed through the UI. This can be prevented using sensitive data filters.

A sensitive data filter is a regular expression to define data that you do not want displayed. When matched, the sensitive portion of the data is encrypted using an MD5 hash. The encrypted data can be compared with earlier versions to determine whether it has changed, while the actual data remains hidden from users.

System communications
Wherever possible, communications between elements of the system use high-grade encryption.

The core of the application manages the discovery and reasoning engines. It consistently interacts with the security engine to ensure user authentication and request authorization so that each action taken by the application can only be triggered from the application itself or by a user through the application UI or command line. External communications between the user and the application can be configured to use HTTPS.

The encryption of communication between the discovery engine (appliance or Windows proxy) and the target depends on the discovery method used. For example, ssh is encrypted, but telnet and rlogin (which might both be disabled) are not. Discovery credentials can be configured to use a user supplied SSH key per credential. These keys and their associated passphrases are stored in the credential vault. It is recommended that SSH keys are always protected with a strong passphrase.

Secure communications
Secure communications between elements of the system use CORBA over TLS (Transport Layer Security) with the following details:

- Protocol: TLSv1.2
- Encryption: AES_256_CBC
- Message hashing: SHA1
- Key Exchange: DHE_RSA (2048)

It is enabled using certificates in the following locations:

- Each appliance (scanning or consolidation)
- Each Windows proxy
- Certificate Authority public certificate on each appliance and proxy
This refers to communications between components of the BMC Atrium Discovery system, not communications between BMC Atrium Discovery and discovery targets, or between the user's web browser and the appliance UI.

For details see Secure deployment.

Apache SSL key passphrases

BMC recommend that you do not passphrase the Apache SSL server key used by the appliance. Doing so requires entry of the passphrase at service start-up, which conflicts with the following operations:

- Resetting configuration of a machine (invoked from the Cluster Management UI and when a machine leaves the cluster)
- Configuring HTTPS (via the UI and possibly when sending configuration to cluster members). A specific issue is that once a passphrase is applied it is no longer possible to restart HTTPS via the UI, without first regenerating the server key.
- Atrium SSO (via the UI and possibly when sending configuration to cluster members)
- Backup/Restore (as SSL keys are restored)

End-user authentication

End-user application authentication is critical to the security of the entire solution. BMC Atrium Discovery supports a number of Web authentication plug-ins and various levels of authentication strength, requiring one of many authentication factors:

- **SSL Client Certificate Verification** - Strong authentication using a public key infrastructure certificate. The client's SSL Certificate is verified by the Web server. The user name is extracted from the certificate and used for authorization via LDAP
- **SSL Certificate Lookup** - The user is authenticated by looking up custom parts of the client's SSL Certificate via LDAP. The certificate is not verified, but it must be valid
- **LDAP Authentication** - The user is authenticated against an LDAP server by entering a username and password
- **Standard Web Authentication** - The user is authenticated as a local user by entering a username and password

Secure export to CMDB

The communication between BMC Atrium Discovery and BMC Atrium CMDB is based on the CMDB API. The encryption that comes with the AR Server is the Standard Encryption 512-bit public key/56-bit DES encryption on the wire. If a customer acquired the higher levels of Remedy Encryption (a separate product), then the customer could obtain either 1024-bit public key/128-bit RC4 or 2048-bit public key/2048-bit RC4 encryption. Communication from BMC Atrium Discovery to the AR Server can be configured to use a single chosen port (ARTCPPORT).
Ports used for System Communication

The following ports are used by the BMC Atrium Discovery and might need to be opened on a firewall for correct operation. These will be required in addition to the ports directly used for Discovery communications.

Incoming Appliance User Interface ports

These ports will need to be open to access the Main UI and CLI for both normal operation and administration of updates. Enabling HTTPS in BMC Atrium Discovery allows any protocols except SSLv1 and SSLv2. The Apache module (mod_ssl) used to provide SSL and TLS support supports the following:

- TLS v1
- TLS v1.1
- TLS v1.2

TLS v1.1 and v1.2 support was added with the openssl-1.0.1e-15.el6 package, this is available in the December 2013 RHEL 6 operating system upgrade.

Note: The Apache module has not been updated so it is not currently possible to explicitly enforce or disable TLS v1.1 or 1.2.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>SSH</td>
<td>Appliance CLI access</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>Main UI Standard</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>Main UI Secure</td>
</tr>
</tbody>
</table>

Incoming / Outgoing Appliance Consolidation port

Consolidation uses TLS communication. The scanning appliances connect to the consolidation appliance using TLS 1.2 to transfer CORBA messages:

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>25032</td>
<td>TLS/CORBA</td>
<td>Consolidation data</td>
</tr>
</tbody>
</table>

Incoming / Outgoing Appliance Clustering ports

Clusters use TLS communication to communicate between the members. All members of the cluster both create outgoing connections to these ports and accept incoming connections on them:

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>25030</td>
<td>TLS/CORBA</td>
<td>Cluster management</td>
</tr>
<tr>
<td>25031</td>
<td>TLS/CORBA</td>
<td>Data store</td>
</tr>
<tr>
<td>25032</td>
<td>TLS/CORBA</td>
<td>Reasoning</td>
</tr>
</tbody>
</table>
Outgoing Appliance Service ports

These ports will be used in general operation. If configured, email alerts will be sent under certain conditions and an SMTP relay needs to be accessible to do this. As part of discovery the current domain names of IPs will be looked up, so access to your DNS infrastructure is required for this to work. It is essential for correct operation of the system that accurate time is kept for timestamps and access to an NTP service might be required for this. If AD/LDAP UI user authentication is desired then access to your AD/LDAP infrastructure is required.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>SMTP</td>
<td>Email Relay</td>
</tr>
<tr>
<td>53</td>
<td>DNS</td>
<td>Domain Name Lookup</td>
</tr>
<tr>
<td>123</td>
<td>NTP</td>
<td>Time Synchronisation</td>
</tr>
<tr>
<td>389</td>
<td>LDAP</td>
<td>LDAP UI User Authentication</td>
</tr>
<tr>
<td>636</td>
<td>LDAPS</td>
<td>Secure LDAP UI User Authentication</td>
</tr>
</tbody>
</table>

Outgoing Appliance CMDB Sync ports

The BMC Atrium CMDB is built on the AR System platform. This uses a portmapper approach to do RPC calls in much the same way that WMI access occurs. As such unless action is taken the ports used will be 111 to contact the portmapper and an ephemeral port will be used for the duration of the connection. You would be advised to arrange your architecture to not have a firewall between the appliance and the CMDB unless your CMDB is set to use a fixed port by setting the ARTCPPORT variable.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARTCPPORT</td>
<td>AR System</td>
<td>CMDB Sync</td>
</tr>
</tbody>
</table>

Incoming Windows Proxy / Outgoing Appliance ports

The Windows proxies listen for incoming connections from appliances. Communication uses TLS 1.2 connections containing CORBA messages. The ports in use can be configured using the Windows Proxy Manager, but the defaults are:

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>4321</td>
<td>TLS/CORBA</td>
<td>Active Directory Proxy</td>
</tr>
<tr>
<td>4322</td>
<td>TLS/CORBA</td>
<td>Obsolete Workgroup Proxy</td>
</tr>
<tr>
<td>4323</td>
<td>TLS/CORBA</td>
<td>Credential Proxy</td>
</tr>
</tbody>
</table>

Outgoing Windows proxy service ports

If an AD Windows proxy is deployed then the Windows Server hosting it must have access to your AD/LDAP infrastructure.
Discovery communications

This section describes communication between the BMC Atrium Discovery appliance, Windows proxies, and discovery targets.

**Base device discovery**

For efficiency, the appliance uses ICMP ping to locate a device. It is possible to use other ping techniques if ICMP Echo is suppressed in your environment. To do so, on the **Administration** tab, scroll down to the Discovery section and click **Discovery Configuration**. On the Scanning section, enable the **Use TCP ACK 'ping' before scanning** and **Use TCP SYN 'ping' before scanning** check boxes, and enter the port numbers in the TCP ports to use for initial scan and UDP ports to use for initial scan fields.

If you do not allow ICMP pings through the firewall and do not enable TCP Ack and Syn pings, you might lose performance. This is because Discovery performs a full "Access Method" nmap port scan to determine whether the host is actually present, which causes delays as Discovery waits for requests to timeout. You must alter the "Ping hosts before scanning" setting to "No" in this situation. If there is a limited range if IPs for which ICMP Echo is suppressed, you can disable the ping behavior for these IPs by using the Exclude ranges from ping. For more information, see **Configuring discovery settings**.

⚠️ To scan networks that do not permit ICMP ping packets, you may set **Use TCP ACK ping before scanning** or **Use TCP SYN ping before scanning** (or both of these) in your discovery settings to **Yes**. If BMC Atrium Discovery pings an IP address where there is no device and some firewall in your environment is configured to respond for that IP address, it may result in reporting a device which does not exist on the network rather than dark space (NoResponse). To avoid this, it is recommended to either alter such firewall configurations or not to enable TCP ACK ping or TCP SYN ping.

If Discovery cannot connect to an endpoint, it uses heuristic techniques to estimate what sort of device is present. These are controlled by options in **Configuring discovery settings**.

Port 4 using TCP and UDP is required if using IP Fingerprinting as Discovery must observe the response from a guaranteed closed port on the endpoint.

Port 4 must be closed on the discovery target, but must be open on any firewall between the appliance and discovery target, so that the response is from the target rather than the firewall. Where this is not the case, the heuristic receives a response from two different TCP/IP stacks,
leading to unpredictable results including the endpoint being classified as a firewall or an
unrecognized device. This can lead BMC Atrium Discovery to skip devices (see
UnsupportedDevice in the DiscoveryAccess page).

The ports listed in the following table are used to determine what device is present.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Closed Port</td>
</tr>
<tr>
<td>21</td>
<td>FTP</td>
</tr>
<tr>
<td>22</td>
<td>SSH</td>
</tr>
<tr>
<td>23</td>
<td>telnet</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
</tr>
<tr>
<td>135</td>
<td>Windows RPC</td>
</tr>
<tr>
<td>161</td>
<td>SNMP</td>
</tr>
<tr>
<td>513</td>
<td>rlogin</td>
</tr>
<tr>
<td>3940</td>
<td>Discovery for z/OS Agent</td>
</tr>
</tbody>
</table>

**SNMP: Ports used for discovery**

The only port required for SNMP discovery is 161 UDP.

**UNIX: Ports used for discovery**

The minimum port required for successful UNIX discovery is just the port associated with the
access methods that you use. For example, if you only use ssh, this will be port 22. The following
table details the assignment for each port number.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>SSH</td>
</tr>
<tr>
<td>23</td>
<td>telnet</td>
</tr>
<tr>
<td>513</td>
<td>rlogin</td>
</tr>
</tbody>
</table>

**Windows: Ports used for discovery**

This section describes the ports that the Windows proxy uses when discovering remote Windows
targets. If you intend to discover hosts behind a firewall, you must open these ports in the firewall.
The ports given are outgoing (from the Windows proxy and the appliance) TCP ports.

**Windows targets and port 135**

The appliance scans port 135 to determine whether the port is open and therefore the target is
likely to be a Windows host. If the port is open, further discovery is performed using the Windows
proxy.

You can disable this behavior. To do so:
1. Choose Administration > Discovery > Discovery Configuration.
2. Select the No option button in the Check port 135 before using Windows access methods field.
   Discovery does not need to detect port 135 as open; it assumes that the target is a Windows host. When you use this setting, all hosts are assumed to be Windows. A UNIX host is scanned unsuccessfully using a Windows proxy before any UNIX access methods are attempted.

WMI

The ports that are used by WMI discovery methods and the corresponding assigned ports are described in the following table.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>135</td>
<td>DCE RPC Endpoint Manager. DCOM Service Control</td>
</tr>
<tr>
<td>1024-1030</td>
<td>Restricted DCOM One of these ports is used after initial negotiation.</td>
</tr>
<tr>
<td>1024-65535</td>
<td>Unrestricted DCOM One of these ports is used after initial negotiation.</td>
</tr>
<tr>
<td>139</td>
<td>Netbios Session Service</td>
</tr>
<tr>
<td>445</td>
<td>Microsoft Directory Services SMB</td>
</tr>
</tbody>
</table>

All WMI communication from BMC Atrium Discovery is sent with Packet Privacy enabled. If the host being discovered does not support Packet Privacy, the flag is ignored and WMI returns the requested information (for example, if you run a version earlier than Windows Server 2003 with Service Pack 1 (SP1)).

⚠️ By default, WMI (DCOM) uses a randomly selected TCP port between 1024 and 65535. To simplify configuration of the firewall, you should restrict this usage if you scan through firewalls. See To set the DCOM Port Range for more information.

Windows NT4 and NT4 style domains (WMI)

TCP 139 is required instead of TCP 445 if you discover NT4 or you authenticate on an NT4-style non-AD Domain (such as a domain run using Samba 3.x or earlier).

TCP 139 is the NetBIOS Session Service. Some versions of Windows (particularly 9x/NT4) run SMB on NetBIOS over TCP using port 139. Newer versions default to running SMB directly over TCP on port 445. Windows XP/2003/Vista/2008 and later and Active Directory networks use SMB directly over TCP 445.
WMI queries from a Windows Server 2008 to a Windows NT4 host fail using the default security settings. On the Windows proxy host, turn off the requirement for 128 bit security in the Network security: Minimum session security for NTLM SSP based (including RPC) clients policy to permit this.

To set the DCOM port range

WMI is based on the Distributed Component Object Model (DCOM) which, by default, uses a randomly selected TCP port between 1024 and 65535 for communications. To make this more efficient for firewalls, the range can be restricted using the following procedure on each Target Host.

⚠️ These settings should be restricted on the target host, not the Windows proxy host.

1. Using a registry editor, create the key
   HKEY_LOCAL_MACHINE\Software\Microsoft\Rpc\Internet
2. Within that key create a REG_MULTI_SZ (Multi-String Value) called Ports.
3. Enter in the port(s) or port range you want to use.
   The Windows proxy uses only one port; however, if the user has other DCOM applications in use on that machine, you might need to enable a larger range.
4. Create a REG_SZ (String Value) called PortsInternetAvailable and give it the value Y.
5. Create a REG_SZ (String Value) called UseInternetPorts and give it the value Y.
6. Restart the computer.

You should also read the relevant Microsoft article about this issue: [How to configure RPC dynamic port allocation to work with firewalls](#)

RemQuery

Although WMI is the standard mechanism for remote system interrogation and management from Microsoft, some operations are not possible using WMI; primarily netstat data in core discovery, and any additional commands run, or file content extraction, via patterns. Without RemQuery we would be unable to determine network connection information for the discovery target, communication between that host and others, and application modelling based on network connections. Additional discovery operations using RemQuery are typically for deeper software discovery, modeling and versioning.

RemQuery is a BMC Atrium Discovery utility that uses the same basic approach as the Microsoft PSEexec tool. It operates in the following way. The proxy copies the RemQuery executable to the ADMIN$ share on the target. Windows Administrator access is required to write to the ADMIN$ share and start the RemQuery service. Once the service is started on the target, the proxy sends its public key to the RemQuery service, which generates an encryption key, encrypts it with the received public key, and then sends it back. The proxy then recovers the encryption key using its private key. From that point, all proxy to RemQuery communication is secured using the encryption
key and an appropriate algorithm, depending on the target system. RemQuery discovery uses AES encryption with a 256 bit key. AES is not supported in Windows 2000 so RemQuery discovery falls back to DES encryption. Windows NT does not support AES or DES so RemQuery discovery is unencrypted. The proxy communicates to the RemQuery service using a named pipe. This pipe is secured so that only an Administrator user can access it.

At the end of the scan, the service is stopped and uninstalled, but the executable is left in the ADMIN$ share. If a copy already exists, and is the same version, it is not copied again. If the copy is of an older version, it is updated.

The ports that are used by RemQuery discovery and the corresponding port assignments are described in the following table.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>139</td>
<td>Netbios Session Service</td>
</tr>
<tr>
<td>445</td>
<td>Microsoft Directory Services SMB</td>
</tr>
</tbody>
</table>

Windows NT4 and NT4 style domains (RemQuery)
TCP 139 is required instead of TCP 445 if you discover NT4 or if you authenticate on an NT4-style non-AD Domain, such as a domain run using Samba 3.x or earlier.

TCP 139 is the NetBIOS Session Service. Some versions of Windows (particularly 9x/NT4) run SMB on NetBIOS over TCP using port 139. Newer versions default to running SMB directly over TCP on port 445. Windows XP/2003/Vista/2008 and later and Active Directory networks use SMB directly over TCP 445.

Mainframe: Ports used for discovery
The only port required for mainframe discovery is 3940 TCP by default. See Discovery Configuration for more information about how to configure this port.

WBEM: Ports used for discovery
The default ports used for WBEM discovery are:

- HTTP port: 5988
- HTTPS ports: 5989

See Discovery Configuration for more information about how to configure these ports.

Ports required for extended discovery
The following sections detail port information for extended discovery types.
J2EE Discovery
The port information used for J2EE discovery is determined in the patterns used to discover the particular J2EE Application Server. If no port information is discovered, then the default port is used. In addition, for full extended discovery, the port for the database that the J2EE Application Server is using is also required. This is dependent on the way that these servers are configured in your organization.

The following table details the default port.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port Assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>7001</td>
<td>JMX</td>
<td>WebLogic</td>
</tr>
</tbody>
</table>

SQL discovery
The port information used for SQL discovery is derived in the patterns used to discover the particular database. This is dependent on the way that databases are configured in your organization.

The following table details the default ports.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port Assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>1521</td>
<td>SQL</td>
<td>Oracle</td>
</tr>
<tr>
<td>1433</td>
<td>SQL</td>
<td>MS SQL</td>
</tr>
<tr>
<td>4100</td>
<td>SQL</td>
<td>Sybase ASE</td>
</tr>
<tr>
<td>3306</td>
<td>SQL</td>
<td>MySQL</td>
</tr>
</tbody>
</table>

VMware ESX/ESXi discovery using vCenter
The ports required for discovery of VMware ESX/ESXi hosts using vCenter are listed in the following table.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port Assignment</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>VMware ESX/ESXi (also on vCenter host)</td>
</tr>
<tr>
<td>902</td>
<td>vSphere API</td>
<td>VMware ESX/ESXi</td>
</tr>
</tbody>
</table>

⚠️ Discovery of vCenter
Discovery of vCenter uses standard host discovery with the creation of a vCenter SI triggered on a discovered vCenter process.

VMware ESX/ESXi discovery using vSphere
The ports required for discovery of VMware ESX/ESXi hosts are listed in the following table.
### Dark space scanning

This release makes some changes to the way that discovery is undertaken in BMC Atrium Discovery. This improves the efficiency of dark space scanning and removes the need for organizations with sparsely populated IP space to perform sweep scans to determine the IP addresses which should be scanned.

Scanning is now undertaken in two distinct parts:

- Pre-scanning
- Scanning

Pre-scanning is the initial stage of discovery. The list of endpoints to be scanned is segmented into blocks of 256, and each block is passed to discovery. The endpoints are then pre-scanned in parallel using the following techniques, if enabled in discovery configuration and the IP address is not part of an exclude range:

- Ping
- TCP ACK ping and/or TCP SYN ping

If pinging is not permitted:

- The endpoints are port scanned, using the TCP/UDP ports configured for initial scans.
- The valid port states setting is used to determine whether a port is open.

If a device is present and desktop discovery is disabled (the default), an SMB query is used to determine whether the endpoint is a Windows desktop. On FIPS enabled appliances, the SMB query is not used and detection of Windows desktops is done later. If the device appears to be a Windows desktop, the WBEM ports are checked. If it responds on the WBEM ports, it continues the scan as the device is most likely an embedded OS on a storage device, otherwise it is classed as a desktop.

The following results are possible for each endpoint:

- Response
- Desktop (only if desktop discovery is disabled)
- No response

Where there is a response, the endpoint is put into the discovery queue for full discovery.
Where the endpoint is determined to be a desktop, the DiscoveryAccess is created with the result
*Skipped (Desktop host discovery has been disabled).*

For endpoints that have no response, the system searches for a DiscoveryAccess with that
endpoint as the last IP address. The following outcomes are possible:

- Not found.
- A DiscoveryAccess is found and it has a dark_space flag set.
- A DiscoveryAccess is found and it does not have a dark_space flag set.

For the first two outcomes, the endpoint is put into a DroppedEndpoints node and no further
investigation of the endpoint is undertaken. For the third, the endpoint is put into the discovery
queue. The DiscoveryAccess is then aged in a similar manner to DDD aging, though it uses
`no_response_count` and `last_response`. If the DiscoveryAccess is successfully aged, then
the dark_space flag is set.

Aging is also applied to DiscoveryAccess nodes with the dark_space flag set, so dark space
endpoints will age out and be removed.

DroppedEndpoints nodes have the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reason</td>
<td>Why the endpoint was dropped. This is one of:</td>
</tr>
<tr>
<td></td>
<td>- Excluded – the endpoint was part of an excluded range</td>
</tr>
<tr>
<td></td>
<td>- Already processing – no longer created in the DiscoveryAccess chain</td>
</tr>
<tr>
<td></td>
<td>- Dark space – this endpoint is dark space</td>
</tr>
<tr>
<td>Start time</td>
<td>When the first endpoint was added.</td>
</tr>
<tr>
<td>End time</td>
<td>When the last endpoint was added.</td>
</tr>
<tr>
<td>Count</td>
<td>How many endpoints are in the list.</td>
</tr>
<tr>
<td>List of IP addresses</td>
<td>A list of all the dropped endpoints.</td>
</tr>
</tbody>
</table>

In previous releases the removed dark space endpoints caused a mismatch in the number of
endpoints linked to discovery runs. As the DroppedEndpoints node records the dark space
endpoints, this no longer occurs. A single DiscoveryRun can be linked to multiple
DroppedEndpoints nodes.

**Consolidation and dark space scanning**

When a version 10.1 consolidator receives data from an earlier version scanner, the consolidator
builds DroppedEndpoints nodes for already processing IP addresses on the consolidator and
skipped optimized IP addresses.
A version 10.1 scanning appliance creates DroppedEndpoints nodes which are consolidated. If the consolidator does not have the endpoint marked as dark space, then an endpoint may be included in a DroppedEndpoints node and have a DiscoveyAccess. This permits a device previously found on an IP address to be aged out in the normal manner.

**RemQuery**

Although WMI is the standard mechanism for remote system interrogation and management from Microsoft, some operations are not possible using WMI; primarily netstat data in core discovery, and any additional commands run, or file content extraction, via patterns. Without RemQuery we would be unable to determine network connection information for the discovery target, communication between that host and others, and application modelling based on network connections. Additional discovery operations using remquery are typically for deeper software discovery, modeling and versioning.

RemQuery is a BMC Atrium Discovery utility that uses the same basic approach as the Microsoft PSExec tool. It operates in the following way. The proxy copies the RemQuery executable to the ADMIN$ share on the target. Windows Administrator access is required to write to the ADMIN$ share and start the RemQuery service. Once the service is started on the target, the proxy sends its public key to the RemQuery service, which generates an encryption key, encrypts it with the received public key, and then sends it back. The proxy then recovers the encryption key using its private key. From that point, all proxy to RemQuery communication is secured using the encryption key and an appropriate algorithm, depending on the target system. RemQuery discovery uses AES encryption with a 256 bit key. AES is not supported in Windows 2000 so RemQuery discovery falls back to DES encryption. Windows NT does not support AES or DES so RemQuery discovery is unencrypted. The proxy communicates to the RemQuery service using a named pipe. This pipe is secured so that only an Administrator user can access it.

The ports that are used by RemQuery discovery and the corresponding port assignments are described in the following table.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>139</td>
<td>Netbios Session Service</td>
</tr>
<tr>
<td>445</td>
<td>Microsoft Directory Services SMB</td>
</tr>
</tbody>
</table>

**Windows NT4 and NT4 style domains (RemQuery)**

TCP 139 is required instead of TCP 445 if you discover NT4 or if you authenticate on an NT4-style non-AD Domain, such as a domain run using Samba 3.x or earlier.

TCP 139 is the NetBIOS Session Service. Some versions of Windows (particularly 9x/NT4) run SMB on NetBIOS over TCP using port 139. Newer versions default to running SMB directly over TCP on port 445. Windows XP/2003/Vista/2008 and later and Active Directory networks use SMB directly over TCP 445.
Firewall Port Summary

For convenience a summary of ports potentially used is listed here. See other references in the Security section for full details of the use of these ports. Ports that might well be customized in your environment are written in *italics*.

Scanning and standalone appliance ports

The following table shows the ports that might be used in scanning (part of a consolidating system) and standalone appliances.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Direction</th>
<th>Use</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Closed Port</td>
<td>Outbound</td>
<td>Base Device Detection</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>21</td>
<td>FTP</td>
<td>Outbound</td>
<td>Base Device Detection</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>22</td>
<td>SSH</td>
<td>Inbound</td>
<td>Appliance CLI access</td>
<td>System communications</td>
</tr>
<tr>
<td>22</td>
<td>SSH</td>
<td>Outbound</td>
<td>UNIX Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>23</td>
<td>telnet</td>
<td>Outbound</td>
<td>UNIX Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>25</td>
<td>SMTP</td>
<td>Outbound</td>
<td>Email Relay</td>
<td>System communications</td>
</tr>
<tr>
<td>53</td>
<td>DNS</td>
<td>Outbound</td>
<td>Domain Name Lookup</td>
<td>System communications</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>Inbound</td>
<td>Main UI Standard</td>
<td>System communications</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>Outbound</td>
<td>Base Device Detection</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>123</td>
<td>NTP</td>
<td>Outbound</td>
<td>Time Synchronization</td>
<td>System communications</td>
</tr>
<tr>
<td>135</td>
<td>DCE RPC Endpoint Manager. DCOM Service Control</td>
<td>Outbound</td>
<td>Windows Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>161</td>
<td>SNMP</td>
<td>Outbound</td>
<td>SNMP Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>389</td>
<td>LDAP</td>
<td>Outbound</td>
<td>LDAP UI User Authentication</td>
<td>System communications</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>Inbound</td>
<td>Main UI Secure vCenter discovery</td>
<td></td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Direction</th>
<th>Use</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>513</td>
<td>rlogin</td>
<td>Outbound</td>
<td>UNIX Discovery</td>
<td>System communications Discovery communications</td>
</tr>
<tr>
<td>636</td>
<td>LDAPS</td>
<td>Outbound</td>
<td>LDAPS UI User Authentication</td>
<td>System communications</td>
</tr>
<tr>
<td>902</td>
<td>vSphere API</td>
<td>Outbound</td>
<td>VMware ESX/ESXi Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>1433</strong></td>
<td>MS SQL</td>
<td>Outbound</td>
<td>MS SQL Extended Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>1521</strong></td>
<td>Oracle SQL</td>
<td>Outbound</td>
<td>Oracle SQL Extended Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>3306</strong></td>
<td>MySQL SQL</td>
<td>Outbound</td>
<td>MySQL SQL Extended Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>3940</strong></td>
<td>Discovery for z/OS Agent</td>
<td>Outbound</td>
<td>Mainframe Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>4100</strong></td>
<td>Sybase SQL</td>
<td>Outbound</td>
<td>Sybase ASE SQL Extended Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>4321</td>
<td>CORBA</td>
<td>Outbound</td>
<td>AD Windows proxy Windows Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>4323</td>
<td>CORBA</td>
<td>Outbound</td>
<td>Credential Windows proxy Windows Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>5988</td>
<td>HTTP</td>
<td>Outbound</td>
<td>WBEM Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>5989</td>
<td>HTTPS</td>
<td>Outbound</td>
<td>WBEM Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>7001</td>
<td>JMX</td>
<td>Outbound</td>
<td>J2EE Extended Discovery</td>
<td>Discovery communications</td>
</tr>
<tr>
<td>25032</td>
<td>CORBA</td>
<td>Outbound</td>
<td>Reasoning communication Scanning appliance. This port is used for consolidation.</td>
<td>Discovery communications</td>
</tr>
<tr>
<td><strong>ARTCPPORT Value</strong></td>
<td>AR System</td>
<td>Outbound</td>
<td>CMDB Sync Standalone appliance. Scanning appliances do not sync to CMDB, this is done from the consolidating appliance.</td>
<td>System communications</td>
</tr>
</tbody>
</table>

### Additional ports used in clustered systems

The following ports are used in clustered systems in addition to those used in scanning and standalone machines.
Consolidation Appliance Ports

Consolidation appliance will not normally do local discovery it will purely consolidate data from scanning appliances.

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Port assignment</th>
<th>Direction</th>
<th>Use</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>SSH</td>
<td>Inbound</td>
<td>Appliance CLI access</td>
<td>System communications</td>
</tr>
<tr>
<td>25</td>
<td>SMTP</td>
<td>Outbound</td>
<td>Email Relay</td>
<td>System communications</td>
</tr>
<tr>
<td>53</td>
<td>DNS</td>
<td>Outbound</td>
<td>Domain Name Lookup</td>
<td>System communications</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>Inbound</td>
<td>Main UI Standard</td>
<td>System communications</td>
</tr>
<tr>
<td>123</td>
<td>NTP</td>
<td>Outbound</td>
<td>Time Synchronization</td>
<td>System communications</td>
</tr>
<tr>
<td>389</td>
<td>LDAP</td>
<td>Outbound</td>
<td>LDAP UI User Authentication</td>
<td>System communications</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>Inbound</td>
<td>Main UI Secure</td>
<td>System communications</td>
</tr>
<tr>
<td>636</td>
<td>LDAPS</td>
<td>Outbound</td>
<td>LDAPS UI User Authentication</td>
<td>System communications</td>
</tr>
<tr>
<td>ARTCPPORT Value</td>
<td>AR System</td>
<td>Outbound</td>
<td>CMDB Sync</td>
<td>System communications</td>
</tr>
<tr>
<td>25032</td>
<td>CORBA</td>
<td>Inbound</td>
<td>Consolidation</td>
<td>Discovery communications</td>
</tr>
</tbody>
</table>

Windows proxy Ports

Proxy port changes in 8.3 SP2

In BMC Atrium Discovery 8.3 SP2 and later, proxies are not limited to the default ports. It is also possible to install multiple proxies of each type on a single host. Consequently, in BMC Atrium Discovery 8.3 SP2 and later you must check the proxy manager to determine which ports the proxies are using. The defaults are the same as previous releases, but installations of additional proxies use incremental ports. You can also use the proxy manager to modify the port that each proxy uses.
## DISA Secure Technical Implementation Guidelines

Any system implemented by the US Department of Defense (DoD) must meet the DISA Secure Technical Implementation Guidelines (STIG). The STIGs are publicly available and may also be implemented by organizations with particular security requirements. Information on Red Hat products compliance with US government certifications can be found on the Red Hat website.

Red Hat Enterprise Linux 6 and the included Apache 2.2 installation can be configured to meet their respective STIG. See the following pages from the Information Assurance Support Environment (IASE):


## Version compliance

The changes made in BMC Atrium Discovery 10.2 to comply with the following STIG rule versions.

<table>
<thead>
<tr>
<th>Component</th>
<th>STIG rule version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Enterprise Linux 6</td>
<td>v1r2</td>
</tr>
<tr>
<td>Apache</td>
<td>v1r2</td>
</tr>
</tbody>
</table>

⚠️ Note

If an issue arises on an appliance that has been customized to meet the STIG, BMC Customer Support may require the issue to be reproduced on an unmodified appliance.
Changes to meet the Red Hat Enterprise Linux 6 and Apache 2.2 STIG in BMC Atrium Discovery 10.2

The following sections detail the changes that have been made in BMC Atrium Discovery 10.1 to comply with STIG rules. Sections are provided listing STIG rules that need to be applied at customer's discretion and those that are not applicable to BMC Atrium Discovery, and the reason for non-applicability.

- STIG rules for RHEL6 addressed with restrictions
- STIG rules for RHEL6 met using compliance script
- STIG rules for RHEL6 not applicable to BMC Atrium Discovery 10.2
- STIG rules for Apache addressed with restrictions
- STIG rules for Apache not applicable to BMC Atrium Discovery 10.2

STIG rules for RHEL6 addressed with restrictions

The following section details the STIG rules for Red Hat Enterprise Linux (RHEL) 6 that have been successfully addressed in BMC Atrium Discovery 10.1 but have restrictions.

⚠️ Note

The table provides links to STIG rule descriptions and details on the STIGviewer website. STIGviewer provides an online, searchable index of Public Domain STIG content, though is not related to DISA. Its content may not be up to date.

<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL-06-000003 V-38463</td>
<td>The system must use a separate file system for /var/log. Note — Only applicable to systems kickstarted in 10.1 or upgraded from systems kickstarted in 10.0 or 9.0.2.</td>
</tr>
<tr>
<td>RHEL-06-000004 V-38467</td>
<td>The system must use a separate file system for the system audit data path. Note — Only applicable to systems kickstarted in 10.1 or upgraded from systems kickstarted in 10.0 or 9.0.2.</td>
</tr>
<tr>
<td>RHEL-06-000349 V-38595</td>
<td>The system must be configured to require the use of a CAC, PIV compliant hardware token, or Alternate Logon Token (ALT) for authentication. Note — By default, CAC, PIV compliant hardware tokens, and Alternate Logon Tokens (ALT) are not supported authentication mechanisms. BMC Atrium Discovery can be configured to use BMC Atrium SSO which connects to the authentication mechanism being used.</td>
</tr>
<tr>
<td>RHEL-06-000518 V-38452</td>
<td>Permissions of files should match the permissions expected in the RPM database. Note — During the application of STIG rule configurations we change the permissions of various files, particularly removing setuid of root, and these files will appear during this test.</td>
</tr>
</tbody>
</table>
STIG rules for RHEL6 met using compliance script

The following section details the STIG rules for Red Hat Enterprise Linux (RHEL) 6 that have been addressed in BMC Atrium Discovery 10.0, but must be enabled to achieve compliance. To enable compliance for all of the rules described in the following tables, run the `tw_stig_control` script as the root user.

The `tw_stig_control` script runs the following scripts which enable STIG compliance for different functional areas of BMC Atrium Discovery. These are:

- `tw_stig_auditing` — the auditing functionality of BMC Atrium Discovery
- `tw_stig_local_env` — the local environment of BMC Atrium Discovery
- `tw_stig_remote_mgmt` — the remote management functionality of BMC Atrium Discovery

Possible lock out

One of the changes made to comply with the STIG is to expire Red Hat Enterprise Linux (RHEL) user passwords every 60 days. After a password has expired, there is a grace period of 35 days during which a user will be allowed to change their password on the first login attempt. After 35 days the user will be completely locked out (this also applies to the root user). Consequently, you should check that the root, tideway and netadmin user passwords have been changed within the last 95 days before applying the STIG scripts described here, or you may be locked out from these accounts (and effectively from the VM itself). The password restrictions are applied by the `tw_stig_local_env` script.

No automatic reversion

There is no automatic facility to revert the changes applied by these scripts.

Auditing creates significant additional logging

The `tw_stig_auditing` script enables auditing on the system. Work has been done in the release to limit the number of privileged commands BMC Atrium Discovery needs to run during discovery but you will need management processes in place to ensure that there is sufficient space for additional logging in the `/var/log` and `/var/log/audit` directories or partitions.
You can choose to run the scripts individually but if you choose not to run a script then the appliance *will not comply* with all of the STIG rules in that functional area.

```
[root@appliance01 bin]# sh tw_stig_control

The following scripts will be run that change the configuration required to satisfy the listed DISA RHEL6 STIG requirements.

./tw_stig_auditing:
RHEL-06-000145, RHEL-06-000148, RHEL-06-000154, RHEL-06-000159
RHEL-06-000160, RHEL-06-000161, RHEL-06-000165, RHEL-06-000167
RHEL-06-000169, RHEL-06-000171, RHEL-06-000173, RHEL-06-000174
RHEL-06-000175, RHEL-06-000176, RHEL-06-000177, RHEL-06-000182
RHEL-06-000183, RHEL-06-000184, RHEL-06-000185, RHEL-06-000186
RHEL-06-000187, RHEL-06-000188, RHEL-06-000189, RHEL-06-000190
RHEL-06-000191, RHEL-06-000192, RHEL-06-000193, RHEL-06-000194
RHEL-06-000195, RHEL-06-000196, RHEL-06-000197, RHEL-06-000198
RHEL-06-000199, RHEL-06-000200, RHEL-06-000201, RHEL-06-000202
RHEL-06-000509, RHEL-06-000525

./tw_stig_local_env:
RHEL-06-000051, RHEL-06-000053, RHEL-06-000056, RHEL-06-000057
RHEL-06-000058, RHEL-06-000059, RHEL-06-000060, RHEL-06-000061
RHEL-06-000069, RHEL-06-000070, RHEL-06-000274, RHEL-06-000299
RHEL-06-000334, RHEL-06-000335, RHEL-06-000356, RHEL-06-000357

./tw_stig_remote_mgmt:
RHEL-06-000230, RHEL-06-000231, RHEL-06-000241, RHEL-06-000319
RHEL-06-000340, RHEL-06-000341

Please note: A reboot is required to complete the configuration changes.

Are you sure you want to perform the configuration changes (yes/no)?
[root@appliance01 bin]#
```

⚠️ **Note**

The table provides links to STIG rule descriptions and details on the [STIGviewer](#) website. STIGviewer provides an online, searchable index of Public Domain STIG content, though is not related to DISA. Its content may not be up to date.

The following table lists the STIG rules concerning the auditing functionality of BMC Atrium Discovery met by running `tw_stig_auditing`. 
<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL-06-000145 V-38628</td>
<td>Auditing must be implemented.</td>
</tr>
<tr>
<td>RHEL-06-000148 V-38631</td>
<td>Auditing must be implemented.</td>
</tr>
<tr>
<td>RHEL-06-000154 V-38632</td>
<td>Auditing must be implemented.</td>
</tr>
<tr>
<td>RHEL-06-000159 V-38636</td>
<td>The system must retain enough rotated audit logs to cover the required log retention period.</td>
</tr>
<tr>
<td>RHEL-06-000160 V-38633</td>
<td>The system must set a maximum audit log file size.</td>
</tr>
<tr>
<td>RHEL-06-000161 V-38634</td>
<td>The system must rotate audit log files that reach the maximum file size.</td>
</tr>
<tr>
<td>RHEL-06-000165 V-38635</td>
<td>The audit system must be configured to audit all attempts to alter system time through adjtimex.</td>
</tr>
<tr>
<td>RHEL-06-000167 V-38622</td>
<td>The audit system must be configured to audit all attempts to alter system time through settimeofday.</td>
</tr>
<tr>
<td>RHEL-06-000169 V-38625</td>
<td>The audit system must be configured to audit all attempts to alter system time through stime.</td>
</tr>
<tr>
<td>RHEL-06-000171 V-38527</td>
<td>The audit system must be configured to audit all attempts to alter system time through clock_settime.</td>
</tr>
<tr>
<td>RHEL-06-000173 V-38530</td>
<td>The audit system must be configured to audit all attempts to alter system time through /etc/ltc/localtime.</td>
</tr>
<tr>
<td>RHEL-06-000174 V-38531</td>
<td>The audit system must be configured to audit account creation and modification.</td>
</tr>
<tr>
<td>RHEL-06-000175 V-38534</td>
<td>The audit system must be configured to audit account creation and modification.</td>
</tr>
<tr>
<td>RHEL-06-000176 V-38536</td>
<td>The audit system must be configured to audit account creation and modification.</td>
</tr>
<tr>
<td>RHEL-06-000177 V-38538</td>
<td>The audit system must be configured to audit account creation and modification.</td>
</tr>
<tr>
<td>RHEL-06-000182 V-38540</td>
<td>The audit system must be configured to audit modifications to the systems network configuration.</td>
</tr>
<tr>
<td>RHEL-06-000183 V-38541</td>
<td>The audit system must be configured to audit modifications to the system’s Mandatory Access Control (MAC) configuration (SELinux).</td>
</tr>
<tr>
<td>RHEL-06-000184 V-38543</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using chmd.</td>
</tr>
<tr>
<td>RHEL-06-000185 V-38545</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using chown.</td>
</tr>
<tr>
<td>RHEL-06-000186 V-38547</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fchmod.</td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RHEL-06-000187 V-38550</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fchmodat.</td>
</tr>
<tr>
<td>RHEL-06-000188 V-38552</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fchown.</td>
</tr>
<tr>
<td>RHEL-06-000189 V-38554</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fchownat.</td>
</tr>
<tr>
<td>RHEL-06-000190 V-38556</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fremovexattr.</td>
</tr>
<tr>
<td>RHEL-06-000191 V-38557</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using fsetxattr.</td>
</tr>
<tr>
<td>RHEL-06-000192 V-38558</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using lchown.</td>
</tr>
<tr>
<td>RHEL-06-000193 V-38559</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using lremovexattr.</td>
</tr>
<tr>
<td>RHEL-06-000194 V-38561</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using lsetxattr.</td>
</tr>
<tr>
<td>RHEL-06-000195 V-38563</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using removexattr.</td>
</tr>
<tr>
<td>RHEL-06-000196 V-38565</td>
<td>The audit system must be configured to audit all discretionary access control permission modifications using setxattr.</td>
</tr>
<tr>
<td>RHEL-06-000197 V-38566</td>
<td>The audit system must be configured to audit failed attempts to access files and programs.</td>
</tr>
<tr>
<td>RHEL-06-000198 V-38567</td>
<td>The audit system must be configured to audit all use of setuid programs.</td>
</tr>
<tr>
<td>RHEL-06-000199 V-38568</td>
<td>The audit system must be configured to audit successful file system mounts.</td>
</tr>
<tr>
<td>RHEL-06-000200 V-38575</td>
<td>The audit system must be configured to audit user deletions of files and programs.</td>
</tr>
<tr>
<td>RHEL-06-000201 V-38578</td>
<td>The audit system must be configured to audit changes to the /etc/sudoers file.</td>
</tr>
<tr>
<td>RHEL-06-000202 V-38580</td>
<td>The audit system must be configured to audit the loading and unloading of dynamic kernel modules.</td>
</tr>
<tr>
<td>RHEL-06-000509 V-38471</td>
<td>The system must forward audit records to the syslog service.</td>
</tr>
<tr>
<td>RHEL-06-000525 V-38438</td>
<td>Auditing must be enabled at boot by setting a kernel parameter.</td>
</tr>
</tbody>
</table>

The following table lists the STIG rules concerning local environment of BMC Atrium Discovery met by running tw_stig_local_env.
<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL-06-000051 V-38477</td>
<td>Users must not be able to change passwords more than once every 24 hours.</td>
</tr>
<tr>
<td>RHEL-06-000053 V-38479</td>
<td>User passwords must be changed at least every 60 days.</td>
</tr>
<tr>
<td>RHEL-06-000056 V-38482</td>
<td>The system must require passwords to contain at least one numeric character.</td>
</tr>
<tr>
<td>RHEL-06-000057 V-38569</td>
<td>The system must require passwords to contain at least one uppercase alphabetic character.</td>
</tr>
<tr>
<td>RHEL-06-000058 V-38570</td>
<td>The system must require passwords to contain at least one special character.</td>
</tr>
<tr>
<td>RHEL-06-000059 V-38571</td>
<td>The system must require passwords to contain at least one lowercase alphabetic character.</td>
</tr>
<tr>
<td>RHEL-06-000060 V-38572</td>
<td>The system must require at least four characters be changed between the old and new passwords during a password change.</td>
</tr>
<tr>
<td>RHEL-06-000061 V-38573</td>
<td>The system must disable accounts after three consecutive unsuccessful login attempts.</td>
</tr>
<tr>
<td>RHEL-06-000069 V-38586</td>
<td>The system must require authentication upon booting into single-user and maintenance modes.</td>
</tr>
<tr>
<td>RHEL-06-000070 V-38588</td>
<td>The system must not permit interactive boot.</td>
</tr>
<tr>
<td>RHEL-06-000274 V-38592</td>
<td>The system must prohibit the reuse of passwords within twenty-four iterations.</td>
</tr>
<tr>
<td>RHEL-06-000334 V-38692</td>
<td>Accounts must be locked upon 35 days of inactivity.</td>
</tr>
<tr>
<td>RHEL-06-000335 V-38694</td>
<td>Accounts must be locked upon 35 days of inactivity.</td>
</tr>
<tr>
<td>RHEL-06-000356 V-38592</td>
<td>The system must require administrator action to unlock an account locked by excessive failed login attempts.</td>
</tr>
<tr>
<td>RHEL-06-000357 V-38501</td>
<td>The system must disable accounts after excessive login failures within a 15-minute interval.</td>
</tr>
</tbody>
</table>

The following table lists the STIG rules concerning the remote management functionality of BMC Atrium Discovery met by running `tw_stig_remote_mgmt`.

<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL-06-000230 V-38608</td>
<td>The SSH daemon must set a timeout interval on idle sessions.</td>
</tr>
<tr>
<td>RHEL-06-000231 V-38610</td>
<td>The SSH daemon must set a timeout count on idle sessions.</td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>RHEL-06-000005 V-38470</td>
<td>The audit system must alert designated staff members when the audit storage volume approaches capacity.</td>
</tr>
<tr>
<td>RHEL-06-000008 V-38476</td>
<td>Vendor-provided cryptographic certificates must be installed to verify the integrity of system software.</td>
</tr>
<tr>
<td>RHEL-06-000011 V-38481</td>
<td>System security patches and updates must be installed and up-to-date.</td>
</tr>
<tr>
<td>RHEL-06-000013 V-38483</td>
<td>The system package management tool must cryptographically verify the authenticity of system software packages during installation.</td>
</tr>
<tr>
<td>RHEL-06-000013 V-38483</td>
<td>The system package management tool must cryptographically verify the authenticity of all software packages during installation.</td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>RHEL-06-000015 V-38487</td>
<td>A file integrity tool must be installed.</td>
</tr>
<tr>
<td>RHEL-06-000016 V-38489</td>
<td>All system command files must be owned by root.</td>
</tr>
<tr>
<td>RHEL-06-000048 V-38472</td>
<td>The Department of Defense (DoD) login banner must be displayed immediately prior to, or as part of, console login prompts.</td>
</tr>
</tbody>
</table>
| RHEL-06-000098 V-38546 | The IPv6 protocol handler must not be bound to the network stack unless needed. | BMC Atrium Discovery supports discovery using IPv6. To disable IPv6:  
1. Edit the `/etc/sysconfig/network` file adding `NETWORKING_IPV6=no`  
2. Ensure that any other IPv6 (`net.ipv6`) entry is commented out.  
3. Edit the `/etc/sysctl.conf` file adding `net.ipv6.conf.all.disable_ipv6 = 1`  
4. Prevent the IPv6 firewall from starting. Enter: `service iptables stop chkconfig ip6tables off` |
<p>| RHEL-06-000136 V-38520 | The operating system must back up audit records on an organization defined frequency onto a different system or media than the system being audited. | This is not applicable for BMC Atrium Discovery out-of-the box because it requires additional services to be configured in the customer's environment. |
| RHEL-06-000137 V-38521 | The operating system must support the requirement to centrally manage the content of audit records generated by organization defined information system components. | This is not applicable for BMC Atrium Discovery out-of-the box because it requires additional services to be configured in the customer's environment. |
| RHEL-06-000240 V-38615 | The SSH daemon must be configured with the Department of Defense (DoD) login banner. | We provide a non-standard post-login banner. |
| RHEL-06-000247 V-38620 | The system clock must be synchronized continuously, or at least daily. | Network time synchronization is not configured by default as customers' preferred time server is not known. |</p>
<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
<th>Reason for non-compliance</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL-06-000248 V-38621</td>
<td>The system clock must be synchronized to an authoritative DoD time source.</td>
<td>Network time synchronization is not configured by default as customers’ preferred time server is not known.</td>
</tr>
<tr>
<td>RHEL-06-000252 V-38625</td>
<td>If the system is using LDAP for authentication or account information, the system must use a TLS connection using FIPS 140-2 approved cryptographic algorithms.</td>
<td>LDAP is not configured by default as customer environments are not known.</td>
</tr>
<tr>
<td>RHEL-06-000253 V-38626</td>
<td>The LDAP client must use a TLS connection using trust certificates signed by the site CA.</td>
<td>LDAP is not configured by default as customer environments are not known.</td>
</tr>
<tr>
<td>RHEL-06-000257 V-38629</td>
<td>The graphical desktop environment must set the idle timeout to no more than 15 minutes.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000258 V-38630</td>
<td>The graphical desktop environment must automatically lock after 15 minutes of inactivity and the system must require user to re-authenticate to unlock the environment.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000259 V-38638</td>
<td>The graphical desktop environment must have automatic lock enabled.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000260 V-38639</td>
<td>The system must display a publicly-viewable pattern during a graphical desktop environment session lock.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000269 V-38652</td>
<td>Remote file systems must be mounted with the nodev&quot; option.&quot;</td>
<td>We do not ship with any remote file systems.</td>
</tr>
<tr>
<td>RHEL-06-000270 V-38654</td>
<td>Remote file systems must be mounted with the nosuid&quot; option.&quot;</td>
<td>We do not ship with any remote file systems.</td>
</tr>
<tr>
<td>RHEL-06-000271 V-38655</td>
<td>The noexec option must be added to removable media partitions.</td>
<td>We do not ship with any remote file systems.</td>
</tr>
<tr>
<td>RHEL-06-000275 V-38659</td>
<td>The operating system must employ cryptographic mechanisms to protect information in storage.</td>
<td>BMC Atrium Discovery data is not encrypted in storage so this rule is not applicable.</td>
</tr>
<tr>
<td></td>
<td>The operating system must protect the confidentiality and integrity of data at rest.</td>
<td>BMC Atrium Discovery data is not encrypted in storage so this rule is not applicable.</td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
<td>Reason for non-compliance</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RHEL-06-000276 V-38661</td>
<td>The operating system must employ cryptographic mechanisms to prevent unauthorized disclosure of data at rest unless otherwise protected by alternative physical measures.</td>
<td>BMC Atrium Discovery data is not encrypted in storage so this rule is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000277 V-38662</td>
<td>The system must use and update a DoD-approved virus scan program.</td>
<td>BMC Atrium Discovery does not use a virus scan program, though it does use tripwire to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>RHEL-06-000284 V-38666</td>
<td>The system must have a host-based intrusion detection tool installed.</td>
<td>BMC Atrium Discovery uses tripwire as a host-based intrusion detection tool.</td>
</tr>
<tr>
<td>RHEL-06-000285 V-38667</td>
<td>The postfix service must be enabled for mail delivery.</td>
<td>Email is not configured or enabled by default in BMC Atrium Discovery.</td>
</tr>
<tr>
<td>RHEL-06-000287 V-38669</td>
<td>X Windows must not be enabled unless required.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000290 V-38674</td>
<td>The xorg-x11-server-common (X Windows) package must not be installed, unless required.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000291 V-38676</td>
<td>The DHCP client must be disabled if not needed.</td>
<td>BMC Atrium Discovery requires a DHCP client, though this must be configured when the appliance is commissioned.</td>
</tr>
<tr>
<td>RHEL-06-000297 V-38685</td>
<td>Temporary accounts must be provisioned with an expiration date.</td>
<td>This is an on site configuration activity so is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000298 V-38690</td>
<td>Emergency accounts must be provisioned with an expiration date.</td>
<td>This is an on site configuration activity so is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000302 V-38695</td>
<td>A file integrity tool must be used at least weekly to check for unauthorized file changes, particularly the addition of unauthorized system libraries or binaries, or for unauthorized modification to authorized system libraries or binaries.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use tripwire to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td></td>
<td>The operating system must employ automated mechanisms, per organization defined frequency, to detect the addition of unauthorized components/devices into the operating system.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use tripwire to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
<td>Reason for non-compliance</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RHEL-06-000303</td>
<td>The operating system must employ automated mechanisms to detect the presence of unauthorized software on organizational information systems and notify designated organizational officials in accordance with the organization defined frequency.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use <code>tripwire</code> to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>V-38698</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RHEL-06-000304</td>
<td>The operating system must provide a near real-time alert when any of the organization defined list of compromise or potential compromise indicators occurs.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use <code>tripwire</code> to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>V-38700</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RHEL-06-000305</td>
<td>The operating system must detect unauthorized changes to software and information.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use <code>tripwire</code> to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>V-38670</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RHEL-06-000306</td>
<td>The operating system must ensure unauthorized, security-relevant configuration changes detected are tracked.</td>
<td>BMC Atrium Discovery does not use AIDE, though it does use <code>tripwire</code> to detect unauthorized changes to the system.</td>
</tr>
<tr>
<td>V-38673</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RHEL-06-000308</td>
<td>Process core dumps must be disabled unless needed.</td>
<td>BMC Atrium Discovery relies on core dumps for debug information. However, if you must disable core dumps, this limits BMC Customer Support's ability to resolve problems. To disable core dumps:</td>
</tr>
<tr>
<td>V-38675</td>
<td></td>
<td>1. Edit the <code>/etc/profile</code> file commenting out the <code>ulimit -S -c unlimited</code> line.</td>
</tr>
<tr>
<td>RHEL-06-000313</td>
<td>The audit system must identify staff members to receive notifications of audit log storage volume capacity issues.</td>
<td>2. Edit the <code>/etc/security/limits.conf</code> file or commenting out or changing the <code>tideway - core unlimited line</code> to <code>tideway - core 0</code>.</td>
</tr>
<tr>
<td>V-38680</td>
<td></td>
<td>3. Edit the <code>/etc/sysctl.conf</code> file commenting out the line <code>kernel.core_pattern = /usr/tideway/cores/</code></td>
</tr>
<tr>
<td>RHEL-06-000321</td>
<td>The system must provide VPN connectivity for communications over untrusted networks.</td>
<td>ADDM does not ship with any VPN tools.</td>
</tr>
<tr>
<td>V-38687</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RHEL-06-000324</td>
<td>A login banner must be displayed immediately prior to, or as part of, graphical desktop environment login prompts.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>V-38688</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
<td>Reason for non-compliance</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RHEL-06-000326 V-38689</td>
<td>The Department of Defense (DoD) login banner must be displayed immediately prior to, or as part of, graphical desktop environment login prompts.</td>
<td>A GUI is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000338 V-38701</td>
<td>The TFTP daemon must operate in secure mode which provides access only to a single directory on the host file system.</td>
<td>TFTP is not installed.</td>
</tr>
<tr>
<td>RHEL-06-000339 V-38702</td>
<td>The FTP daemon must be configured for logging or verbose mode.</td>
<td>No FTP daemons are installed.</td>
</tr>
<tr>
<td>RHEL-06-000341 V-38653</td>
<td>The snmpd service must not use a default password.</td>
<td>The snmpd service is disabled by default. If you enable the snmpd service, you must change the password from the default to be STIG compliant.</td>
</tr>
<tr>
<td>RHEL-06-000348 V-38599</td>
<td>The FTPS/FTP service on the system must be configured with the Department of Defense (DoD) login banner.</td>
<td>No FTP daemons are installed.</td>
</tr>
<tr>
<td>RHEL-06-000349 V-38595</td>
<td>The system must be configured to require the use of a CAC, PIV compliant hardware token, or Alternate Logon Token (ALT) for authentication.</td>
<td>CAC, PIV compliant hardware tokens, and Alternate Logon Tokens (ALT) are not supported authentication mechanisms.</td>
</tr>
<tr>
<td>RHEL-06-000504 V-38488</td>
<td>The operating system must conduct backups of user-level information contained in the operating system per organization defined frequency to conduct backups consistent with recovery time and recovery point objectives.</td>
<td>This is an on site configuration activity so is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000505 V-38486</td>
<td>The operating system must conduct backups of system-level information contained in the information system per organization defined frequency to conduct backups that are consistent with recovery time and recovery point objectives.</td>
<td>This is an on site configuration activity so is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000515 V-38460</td>
<td>The NFS server must not have the all_squash option enabled.</td>
<td>Not applicable as NFS is not installed on a BMC Atrium Discovery appliance.</td>
</tr>
<tr>
<td>RHEL-06-000521 V-38446</td>
<td>The mail system must forward all mail for root to one or more system administrators.</td>
<td>Mail forwarding is an on site configuration.</td>
</tr>
<tr>
<td>RHEL-06-000524 V-38439</td>
<td>The system must provide automated support for account management functions.</td>
<td>This is an on site configuration activity so is not applicable.</td>
</tr>
<tr>
<td>RHEL-06-000020</td>
<td>The system must use a Linux Security Module configured to enforce limits on system services.</td>
<td></td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
<td>Reason for non-compliance</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>RHEL-06-000228</td>
<td>The SSH daemon must be configured to use only the SSHv2 protocol.</td>
<td>BMC Atrium Discovery does not use the default Red Hat login prompt. It will not be replaced with a DoD banner. See RHEL-06-000073.</td>
</tr>
<tr>
<td>RHEL-06-000286</td>
<td>The x86 CTRL-ALT-DELETE key sequence must be disabled.</td>
<td>The BMC Atrium Discovery appliance was configured so that only a log message is generated when the CTRL-ALT-DELETE key sequence is pressed.</td>
</tr>
<tr>
<td>RHEL-06-000507</td>
<td>The operating system, upon successful logon, must display to the user the date and time of the last logon or access via ssh.</td>
<td>In /etc/ssh/sshd_config on BMC Atrium Discovery appliance, the PrintLastLog value is not defined explicitly. The default value is used instead, which satisfies the STIG.</td>
</tr>
</tbody>
</table>

**STIG rules for Apache addressed with restrictions**

There are no restrictions on the the STIG rules for Apache that have been successfully addressed in BMC Atrium Discovery 10.1.

**STIG rules for Apache not applicable to BMC Atrium Discovery 10.2**

The following section details the STIG rules for Apache that are not applicable to BMC Atrium Discovery 10.2 and gives a brief explanation of reasons.

> **Note**

The table provides links to STIG rule descriptions and details on the STIGviewer website. STIGviewer provides an online, searchable index of Public Domain STIG content, though is not related to DISA. Its content may not be up to date.

<table>
<thead>
<tr>
<th>Rule number</th>
<th>Description</th>
<th>Reason for non-compliance</th>
</tr>
</thead>
<tbody>
<tr>
<td>V-2232</td>
<td>The web server password(s) must be entrusted to the SA or Web Manager.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-2234</td>
<td>Public web server resources must not be shared with private assets.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-2236</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rule number</td>
<td>Description</td>
<td>Reason for non-compliance</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>V-2242</td>
<td>A public web server must be isolated in the enclave.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-2243</td>
<td>A private web server must be located on a separate controlled access subnet.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-2248</td>
<td>Web administration tools must be restricted to the web manager and the web manager's designees.</td>
<td>Apache configuration files are 644, owned by root. BMC Atrium Discovery is built as an application appliance rather than a generic user system. Apache forms part of the application stack which is managed by the tideway user. Additional users created once deployed are the customer's responsibility.</td>
</tr>
<tr>
<td>V-2251</td>
<td>&quot;All utility programs, not necessary for operations, must be removed or disabled.&quot;</td>
<td>We limit the number of packages installed on the appliance to those required for functionality or dependency.</td>
</tr>
<tr>
<td>V-2265</td>
<td>Java software installed on the web server must be limited to class files and the JAVA virtual machine.</td>
<td>BMC Atrium Discovery requires additional features available in full JDK installations.</td>
</tr>
<tr>
<td>V-6485</td>
<td>Web server content and configuration files must be part of a routine backup program.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-6577</td>
<td>A web server must be segregated from other services.</td>
<td>BMC Atrium Discovery is not a co-hosted service, it is built as an application appliance rather than a generic user system. Apache forms part of the application stack which is managed by the tideway user.</td>
</tr>
<tr>
<td>V-13620</td>
<td>A private web server's list of CAs in a trust hierarchy must lead to an authorized DoD PKI Root CA.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-13621</td>
<td>&quot;All web server documentation, sample code, example applications, and tutorials must be removed from a production web server.&quot;</td>
<td>We install man pages for all our httpd* packages. These pages are owned by root and 644. The httpd documentation is retained for supportability.</td>
</tr>
<tr>
<td>V-13672</td>
<td>The private web server must use an approved DoD certificate validation process.</td>
<td>Onsite Configuration. A customer deployment consideration.</td>
</tr>
<tr>
<td>V-13732</td>
<td>The &quot;-FollowSymLinks&quot; setting must be disabled.</td>
<td>BMC Atrium Discovery requires symlinks to be available. We have limited this to only the directories that require the ability to follow symlinks.</td>
</tr>
<tr>
<td>V-13736</td>
<td>The HTTP request message body size must be limited.</td>
<td>BMC Atrium Discovery requires a large size for request bodies particularly for upgrade upload.</td>
</tr>
<tr>
<td>V-26322</td>
<td>The score board file must be properly secured.</td>
<td>ScoreBoard file does not exist on the system but is managed in memory.</td>
</tr>
<tr>
<td>V-26326</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Rule number | Description                                                                 | Reason for non-compliance
--- | --- | ---
1. | The web server must be configured to listen on a specific IP address and port. | Onsite Configuration. A customer deployment should modify the Apache configuration files to use the required IP address. This is overwritten by BMC Atrium Discovery upgrades and must be reapplied after the upgrade.

## Running in FIPS compliant mode

The Federal Information Processing Standard (FIPS) Publication 140-2, is a computer security standard, developed by a U.S. Government and industry working group to validate the quality of cryptographic modules.

FIPS Publication 140-2 can be downloaded from the National Institute of Standards and Technology (NIST) web site.

### BMC Atrium Discovery and FIPS

Enabling FIPS mode ensures that BMC Atrium Discovery uses only FIPS compliant cryptographic algorithms and FIPS compliant keys, though some functionality is not supported in FIPS mode, such as using SMB file systems for export or backup. FIPS mode requires that you provide the FIPS compliant SSL keys.

When not running in FIPS mode, BMC Atrium Discovery still uses FIPS compliant cryptographic algorithms where possible.

To fully enable strict FIPS compliance, you must install BMC Atrium Discovery from the kickstart DVD replacing the install or custom options with installfips or customfips. You must also enable NSS after enabling FIPS. For more information on the FIPS compliance, see the Red Hat website.

FIPS mode is only available for BMC Atrium Discovery 9.0 and later versions running on Red Hat Enterprise Linux 6. If you have upgraded an appliance from an earlier version, FIPS mode is not available.

You cannot mount a Windows share from a FIPS enabled appliance. The mount operation fails and an error message is written to syslog.

### Enabling FIPS mode on the appliance

To enable FIPS mode, you must run a script. The script modifies the boot configuration file and regenerates the boot-time kernel. This requires a reboot. Any modifications that have been made to these components may conflict with FIPS mode configuration or have untoward effects.

To enable FIPS mode on the appliance:

1. Login to the appliance command line as the root user.
2. Run the `tw_fips_control` script with the `--enable` option.

```
[root@appliance01 ~]# /usr/tideway/bin/tw_fips_control --enable
```

This script will enable or disable FIPS mode on your ADDM appliance. The script must be run as the root user and FIPS mode is only supported on Red Hat Enterprise Linux 6 based ADDM appliances.

Please note: To enable FIPS mode the script will modify the system's boot configuration files (GRUB) and regenerate the boot-time kernel. Any manual modifications made to these components may conflict with FIPS mode configuration or have untoward effects.

A reboot is required if the current kernel mode needs to change. The script will notify the user if this is the case.

Do you want to continue to enable FIPS mode (yes/no)? yes

Starting FIPS mode configuration.

Gathering current state of the system.

Enable FIPS mode in grub configuration file.

```
WARNING: The default SSL keys shipped with ADDM are NOT FIPS compliant.
You MUST install your own FIPS compliant SSL keys if you have not already done so. Failure to install FIPS compliant keys will mean that ADDM services will not start.
```

Configuration complete. Please reboot to enable FIPS mode.

```
[root@appliance01 ~]#
```

Disabling FIPS mode on the appliance is accomplished by running the `tw_fips_control` script with the `--disable` option. The script modifies the boot configuration file and regenerates the boot-time kernel. This requires a reboot. You do not need to replace SSL keys after disabling FIPS mode.

**Enabling FIPS mode on the proxy**

When installing a proxy the installation detects whether the Windows host is running in FIPS mode. If the host is running in FIPS mode, and you are upgrading from a very old Windows proxy version, you must replace the SSL key before running the proxy. The installer displays a dialog stating this when you install a proxy onto a FIPS enabled host.
For information on using Windows in FIPS mode, see this Microsoft knowledgebase article.

**Enabling NSS**

The mod_ssl Apache library used in BMC Atrium Discovery is not FIPS 140-2 compliant. To achieve strict compliance you must modify the appliance to use `mod_nss` rather than `mod_ssl`. See the Mozilla website for an overview of NSS.

### Restrictions

- The HTTPS administration tools provided in the BMC Atrium Discovery UI should not be used after making configuration changes.
- Any changes made to `ssl.conf` during upgrade are not made to `nss.conf`.
- BMC Atrium Discovery upgrade may overwrite changes/re-enable SSL.

The changes that you must make to enable NSS are:

- Import the correctly formatted certificate and key (pkcs12) into the NSS database. The database is created by this process.
- Ensure permissions are correct on the database
- Configure `nss.conf` and disable `ssl.conf`

**Preparing to configure NSS**

Before you can configure NSS, you must have keys in the pkcs12 format suitable for importing into the NSS database. Depending whether you generate keys using BMC Atrium Discovery, or you already have your own keys you must perform one or both of the following preparatory steps:

- Generating self signed keys
- Converting your keys

If you have your own keys and they are already in pkcs12 format, proceed directly to the To configure NSS procedure.
Generating self signed keys

If you do not have your own certificates and keys:

1. Use the BMC Atrium Discovery HTTPS Configuration UI to generate self signed keys.
2. Start the Converting your keys procedure.

Converting your keys

If you have your own keys and they are not in pkcs12 format, or if you have just generated keys using the HTTPS Configuration UI:

1. Stop the services (tideway, cluster, omniNames, httpd, appliance).
2. Convert your server certificate and key (server.crt and server.key).

```
[root@localhost test]# openssl pkcs12 -export -in /etc/httpd/conf/ssl.crt/server.crt -inkey /etc/httpd/conf/ssl.key/server.key -out server.p12 -name "ADDM-Server-Cert" -passout pass:'Pa55wud!' 
```

To configure NSS

1. As the root user, create the NSS database directory.

```
[root@localhost test]# mkdir /usr/tideway/nssdb
```

2. Import the correctly formatted server certificate and key ensuring that you specify a password.

⚠️ This password will be used for NSS DB tokens (including NSS FIPS 140-2 Certificate DB) so ensure that its complexity meets requirements. The password requested here is not the password provided during the pkcs12 file generation but the password that will be the "NSS Certificate DB" and "NSS FIPS 140-2 Certificate DB" tokens in the NSS database. A password must be set or httpd will fail to start if NSSFips is set to on.

ℹ️ This example uses /usr/tideway/nssdb to store the NSS database. The default NSS database location is /etc/httpd/alias and files are created there by the installation of the NSS RPMs supplied in the RHEL distribution. This database may have been generated with out of date binaries. To use the default location, all files under /etc/httpd/alias should be removed.
[root@localhost test]# pk12util -i server.p12 -d /usr/tideway/nssdb -W 'Pa55wud!' Enter a password which will be used to encrypt your keys. The password should be at least 8 characters long, and should contain at least one non-alphabetic character.

Enter new password:
Re-enter password:
pk12util: PKCS12 IMPORT SUCCESSFUL

3. Confirm the import using the following command:

[root@localhost test]# certutil -L -d /usr/tideway/nssdb
Certificate Nickname                                                                 Trust Attributes
ADDM-Server-Cert                                                                  SSL,S/MIME,JAR/XPI

4. Enable FIPS token in the NSS DB

[root@localhost test]# modutil -fips true -dbdir /usr/tideway/nssdb
WARNING: Performing this operation while the browser is running could cause corruption of your security databases. If the browser is currently running, you should exit browser before continuing this operation. Type 'q <enter>' to abort, or <enter> to continue:

FIPS mode enabled.

5. Set the permissions on the NSS DB so that only the apache and root users can read or write the files

[root@localhost tideway]# chown -R apache:apache /usr/tideway/nssdb

6. Create /etc/httpd/conf.d/nss.conf. This configuration replaces mod_ssl with mod_nss and uses port 443 to avoid modifications to the firewall.

# This is the Apache server configuration file providing SSL support using the mod_nss plugin. It contains the configuration directives to instruct the server how to serve pages over an https connection.
6. LoadModule nss_module modules/libmodnss.so

# When we also provide SSL we have to listen to the
# standard HTTP port (see above) and to the HTTPS port
# Using port 443 to avoid firewall modifications
Listen 443

7. NSSPassPhraseDialog builtin prompts for the database password when the httpd service is started.

## SSL Global Context
## All SSL configuration in this context applies both to
## the main server and all SSL-enabled virtual hosts.
##
## Some MIME-types for downloading Certificates and CRLs
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl .crl

# Pass Phrase Dialog:
# Configure the pass phrase gathering process.
# The filtering dialog program ('builtin' is a internal
terminal dialog) has to provide the pass phrase on stdout.
NSSPassPhraseDialog builtin

8. This can be replaced with the location of a file, additional details below.

# Pass Phrase Helper:
# This helper program stores the token password pins between
# restarts of Apache.
NSSPassPhraseHelper /usr/sbin/nss_pcache

9. The default for NSSSessionCacheTimeout is NSSSessionCacheTimeout 100. This value is set to 300, to match the BMC Atrium Discovery web server’s ssl configuration.

# Configure the SSL Session Cache.
# NSSSessionCacheSize is the number of entries in the cache.
# NSSSessionCacheTimeout is the SSL2 session timeout (in seconds).
# NSSSession3CacheTimeout is the SSL3/TLS session timeout (in seconds).
NSSSessionCacheSize 10000
NSSSessionCacheTimeout 300
NSSSession3CacheTimeout 86400
The **NSSRandomSeed** in the example is set to use `/dev/urandom` to avoid entropy blocking issues. The use of `/dev/urandom` matches the default SSL configuration when using the BMC Atrium Discovery HTTPS generation process.

```plaintext
# Pseudo Random Number Generator (PRNG):
NSSRandomSeed startup file:/dev/random 512
NSSRandomSeed startup file:/dev/urandom 512

# TLS Negotiation configuration under RFC 5746
# Only renegotiate if the peer's hello bears the TLS renegotiation_info extension. Default off.
NSSRenegotiation off

# Peer must send Signaling Cipher Suite Value (SCSV) or Renegotiation Info (RI) extension in ALL handshakes. Default: off
NSSRequireSafeNegotiation off
```

10. Start of the Virtual Host configuration:

```plaintext
## SSL Virtual Host Context

<VirtualHost _default_:443>
ErrorLog logs/ssl_error_log
LogFormat "%h %l %u %t %m %U %H %>s %b"
TransferLog logs/ssl_access_log
LogLevel warn
ServerName localhost.localdomain

# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
NSSEngine on

# Enable FIPS
NSSFips on

# SSL Cipher Suite:
# List the ciphers that the client is permitted to negotiate.
NSSCipherSuite +rsa_3des_sha,-rsa_des_sha,-rsa_rc4_40_md5,-rsa_rc4_40_md5,-rsa_null_md5,-rsa_null_sha,+fips_3des_sha,-fips_des_sha,-fortezza,-fortezza_rc4_128_sha,-fortezza_null,-rsa_des_56_sha,-rsa_rc4_56_sha,+rsa_aes_128_sha,+rsa_aes_256_sha

# Supported Protocols
NSSProtocol TLSv1.1

# SSL Certificate Nickname:
# The nickname of the RSA server certificate you are going to use.
```
11. **NSSVerifyClient** set to *none*. This depends on your environment.

12. **NSSEnforceValidCerts** set to off to allow self-signed certificates to be present in database.

```bash
# Redirect all UI requests to the cluster management status display if it is active
RewriteCond /usr/tideway/var/cluster-manager-redirect-marker -f
RewriteRule ^/ui($|/.*) /status_display/status-display.html?name=ClusterManager [R,L]

# Redirect all UI requests to the ADDM upgrade status display if it is active
RewriteCond /usr/tideway/var/upgrade-redirect-marker -f
RewriteRule ^/ui($|/.*) /status_display/status-display.html?name=upgrade [R,L]

# Redirect all UI requests to the tw_model_wipe status display if it is active
RewriteCond /usr/tideway/var/model-wipe-redirect-marker -f
RewriteRule ^/ui($|/.*) /status_display/status-display.html?name=model_wipe [R,L]

# Redirect all UI requests if we have been ejected from the cluster
RewriteCond /usr/tideway/var/ejected-redirect-marker -f
RewriteRule ^/ui($|/.*) /ejected-from-cluster.html [PT,L]

# Redirect all UI requests to the backup status display if it is active
RewriteCond /usr/tideway/var/backup-redirect-marker -f
RewriteRule ^/ui($|/.*) /status_display/status-display.html?name=backup [R,L]

# Redirect all UI requests to the disk configuration status display if it is active
RewriteCond /usr/tideway/var/disk-configuration-redirect-marker -f
RewriteRule ^/ui($|/.*) /status_display/status-display.html?name=disk_configuration [R,L]

# Redirect all UI requests if we've shutdown because of disk space
RewriteCond /usr/tideway/var/shutdown-disk-space-marker -f
RewriteRule ^/ui($|/.*) /shutdown-disk-space.html [PT,L]

# Otherwise, redirect all UI requests if services are stopped
```
13. Set permissions of the `/etc/httpd/conf.d/nss.conf` file. The file needs to be readable by the `apache` user.

14. Move out or delete `/etc/httpd/conf.d/ssl.conf`.

15. Start `httpd` to confirm config. This requires the NSS database password.

16. Restart all services.

**Additional configuration (optional)**

The following configuration steps are optional:

- Create and use a password file
- Enabling port 80 to port 443 redirection

**Create and use a password file**

Rather than being prompted for a password when starting `httpd`, you can use a password file. To do this:

1. Create a file with the following contents:

   ```
   Internal:<password>
   NSS FIPS 140-2 Certificate DB:<password>
   ```

2. The password is the password of the tokens in the NSS database. The `NSSPassPhraseDialog` directive should be updated to the format shown below, referring to the file created:

   ```
   NSSPassPhraseDialog file:/path/to/file
   ```

   The `apache` user needs to read this file but permissions should be as tight as possible, for example `apache:apache 600`.

**Enabling port 80 to port 443 redirection**

To enable port 80 to port 443 redirection, add the following to the top of the `nss.conf` file.
PCI Data Security Standard compliance

What is PCI DSS compliance?

The Payment Card Industry (PCI) Data Security Standard (DSS) is a set of requirements designed to ensure that all companies that process, store or transmit credit card information maintain a secure environment.

Is BMC Atrium Discovery PCI DSS compliant?

PCI DSS compliance refers to an organization and its environment and processes rather than a discrete software application. Typically, the information security team of the organization is responsible for assessing and validating PCI DSS compliance. BMC Atrium Discovery is regularly deployed in organizations which are PCI compliant but does not carry a certification of PCI compliance.

If a PCI DSS compliance assessment for BMC Atrium Discovery in your site is required, your information security team should use the information in the Security pages for the assessment. If the information security team has questions that have not been answered in the Security pages, they should contact BMC Customer Support with specific questions.
Planning

This section provides important information for planning the deployment of BMC Atrium Discovery.

- Sizing and scalability considerations
- Performance benchmarks and tuning
- System requirements
- Licensing entitlement

Sizing and scalability considerations

Based on typical deployment scenarios, the following guidelines contain information about the recommended hardware specification requirements and configurations for your environments.

- Hardware requirements
- Sizing - initial guidelines

Hardware requirements

This page provides information on recommended specification levels for machines running standalone or as part of a cluster. It contains the following sections:

- What hardware should I run BMC Atrium Discovery on?
  - Minimum specification
  - A “maximum specification”
  - Recommended specification
  - How does Big Discovery change things?
  - How many machines do I need to scan my environment?
  - Fault tolerance
  - Previous recommendations

What hardware should I run BMC Atrium Discovery on?

BMC Atrium Discovery runs well on a variety of platform specifications. The more powerful the hardware on which BMC Atrium Discovery runs, or the proportion of the hardware allocated to it when deployed as a virtual machine, it:

- scans faster
- is more responsive to logged in users
- can store more data

Big Discovery clustering enables you to use two or more machines to increase scanning speed, system responsiveness, and total amount of data that can be handled. Big Discovery can help you reach levels of scale not achievable with current hardware on a single machine, or allow you to reach the same scale with cheaper hardware.
Minimum specification

BMC Atrium Discovery, whether running standalone or as part of a cluster, will run on the following minimum specification hardware:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>4 GB</td>
</tr>
<tr>
<td>Cores</td>
<td>2</td>
</tr>
<tr>
<td>CPU speed</td>
<td>Any 64 bit processor</td>
</tr>
<tr>
<td>Disks</td>
<td>50 GB</td>
</tr>
<tr>
<td>Network</td>
<td>100 Mb Ethernet</td>
</tr>
</tbody>
</table>

This is still only a guideline, not a hard limit. Attempting to run BMC Atrium Discovery on lower specification hardware than this will result in very poor performance. Even on hardware matching this specification, BMC Atrium Discovery will not run particularly well – discovery speeds will be quite low, and the 50 GB disk will soon fill.

A “maximum specification”

Increasing the amount of RAM, CPU speed, cores, and disk and network speeds steadily increases the performance of BMC Atrium Discovery. It is best to increase specifications across the board, otherwise one element will become the bottleneck. For example, there is little to be gained by putting two modern 12-core Xeons in a machine with 4 GB RAM. Instead, divide the investment between CPU, RAM, and disk.

There is a point, though, where BMC Atrium Discovery no longer makes good use of the hardware. We have improved the way in which BMC Atrium Discovery uses powerful hardware in each of the last few releases (10.x, 9.x, and 8.3) and continue to improve it, but for now there is not much to gain by going beyond 12 or so cores. More RAM will always help, but beyond around 64 GB it may not be worth the additional expense.

Recommended specification

Considering the hardware available today, the following specification range gives a good performance/price ratio:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>8 - 32 GB</td>
</tr>
<tr>
<td>Cores</td>
<td>4 - 12</td>
</tr>
<tr>
<td>CPU speed</td>
<td>2 x 3.2 GHz Xeon E3</td>
</tr>
<tr>
<td>Disks</td>
<td>See below.</td>
</tr>
<tr>
<td>Network</td>
<td>Gigabit Ethernet</td>
</tr>
</tbody>
</table>

Disk size should be chosen according to how much you need to scan and how much directly discovered data history you want to retain. Disks should be local (not SAN) and you should have access to multiple spindles – at least two. Having two spindles enables you to separate the datastore data from the datastore transaction logs, which greatly improves performance.
From BMC Atrium Discovery 10.2 CMDB synchronization supports multiple CMDB connections. Disk space consumed for each device (for example a host node) for each connection, is typically 200kB, though this may vary depending on your data.

The CPU given is an example. Some kind of mid-range modern CPU is ideal.

8 GB of RAM is fine, but you are likely to get considerably better performance with 16 GB.

**Memory and swap**

Memory and swap usage depends on the nature of the discovery being performed, with Mainframe and VMware (vCenter/vSphere) devices requiring more than basic UNIX and Windows devices. You can determine whether additional memory is needed in your appliance by monitoring swap usage.

The disk configuration utility uses the following calculation to determine the best swap size.

- Where the amount of memory is less than 16 GB swap size is set at double the memory size.
- Where the amount of memory is between 16 and 32 GB swap is set at 32 GB.
- Where the amount of memory exceeds 32 GB swap is set to equal the memory.

The recommended figures for swap can be exceeded, there is no harm in doing so. It might prove simpler to configure the higher quantity of swap than to extend an existing swap partition as this will allow the RAM demand to be derived as above. All virtual appliances are initially configured with 8 GB swap.

**How does Big Discovery change things?**

Big Discovery enables you to scan more, quicker, and with modest hardware. For example, instead of deploying BMC Atrium Discovery 9 on a 64 GB, 16-core machine, you might install BMC Atrium Discovery 10 on a cluster of four 16 GB, 4-core machines, and the performance is likely to actually be slightly better.

You can increase scanning speed by increasing the number of machines in your cluster, and increasing the specification of each machine. It is your choice how you go about this. You could upgrade all the machines in your 3-machine cluster from 8 GB RAM to 16 GB RAM, or you could add another 8 GB machine to the cluster to make a 4-machine cluster. Both increase performance; the one you decide on will depend on your business policies.

**How many machines do I need to scan my environment?**

It is very difficult to say how much hardware you need to scan a given environment. The performance of BMC Atrium Discovery depends on many factors:

- Performance of your network.
- The environment being scanned
  - dark space vs non dark space
- size of machines being scanned
- amount of software running on those machines
- and many other considerations
- Patterns you have loaded.
- and so on

However, as a rough guideline, you can scan an environment of 10,000 to 15,000 machines every day with a single BMC Atrium Discovery version 10 appliance running on reasonable hardware.

Fault tolerance

Should you turn fault tolerance on? Here are the reasons for and against:

For

- Enabling fault tolerance means a machine in your cluster can fail (see below) and you suffer no data loss and very limited service interruption (in the order of minutes).

Against

- Enabling fault tolerance slows scanning speed.
- Enabling fault tolerance means you can store less data.

A "failing" machine means:

- Hardware failure
- Network failure
- Power loss
- Crash

With fault tolerance disabled, and assuming each machine in a cluster is identical, each machine added to a cluster increases its scanning speed. With fault tolerance enabled, the slow down in scanning speed is evident.

<table>
<thead>
<tr>
<th>Cluster size</th>
<th>Number of hosts that can be scanned, scanning once per day</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fault tolerance disabled</td>
</tr>
<tr>
<td>1</td>
<td>10,000 – 15,000</td>
</tr>
<tr>
<td>2</td>
<td>15,000 – 25,000</td>
</tr>
<tr>
<td>3</td>
<td>20,000 – 35,000</td>
</tr>
<tr>
<td>4</td>
<td>30,000 – 45,000</td>
</tr>
<tr>
<td>6</td>
<td>40,000 – 65,000</td>
</tr>
<tr>
<td>8</td>
<td>50,000 – 80,000</td>
</tr>
</tbody>
</table>
Consolidation does not change these figures – they apply whether you are scanning or consolidating. So, to consolidate 40,000 hosts a day, you need a cluster of 6 machines – just as if you want to scan 40,000 hosts a day.

Note that these figures are for a cluster running on *reasonable* hardware. It is possible to get the same performance with fewer machines by using more powerful hardware.

**Previous recommendations**

In previous releases, sizing guidelines were provided by defining classes of appliance deployment. The introduction of Big Discovery means that these classes are no longer valid for clustered deployments. They do still have some relevance for single appliance deployments, but where scale is required, just add hardware!

The appliance specification page in the UI retains the classes for standalone deployments, but not in clusters. We are reviewing BMC Atrium Discovery performance as version 10 is deployed, and might extend our recommendations based on our customer’s real experiences.

For more information on the previous recommendations, see [Sizing guidelines](#) in the version 9.0 documentation.

### Sizing - initial guidelines

The following initial guidelines are based on typical deployments of standalone systems in the field, and are intended to serve only as recommended configurations for your environment.

This section defines four "classes" of appliance deployment that broadly follow how BMC Atrium Discovery is deployed in the field. They are differentiated by how many Operating System Instances (OSIs) that are being scanned by BMC Atrium Discovery. The names given to these classes are of use only in this document and do not relate to the various editions that BMC Atrium Discovery is available in.

The classes are:

- **Proof of Concept**. Small, time-limited test deployments of BMC Atrium Discovery, scanning up to 150 OSIs.
- **Baseline**. A typical baseline as offered by BMC. Scanning up to 500 OSIs.
- **Datacentre**. A typical large scale deployment. Scanning up to 5000 OSIs.
- **Consolidated Enterprise**. Enterprise scale deployments, typically a Consolidation Appliance taking feeds from many Scanning Appliances. Typically scanning or consolidating of the order of 20000 OSIs or more. At these levels, a weekly scanning or focused scanning strategy may need to be adopted.
The Proof of Concept class has minimal storage allowance as they are only intended for a limited period of scanning such as a week long trial. For longer periods or a continuously used development or UAT system, the Baseline class is the minimum recommended.

Memory and swap considerations

The recommended figures for memory provide a good level of performance in typical scenarios. The upper level should not be considered a limit, as BMC Atrium Discovery will make use of all available memory. You can determine whether additional memory is needed in your appliance by monitoring swap usage.

The recommended figures for swap can be exceeded; there is no harm in doing so, and having more swap can mean more RAM is available for buffers and caches, improving performance. All virtual appliances are initially configured with 8 GB swap.

Memory and swap usage

Memory and swap usage depends on the nature of the discovery being performed, with Mainframe and VMware (vCenter/vSphere) devices requiring more than basic UNIX and Windows devices.

Where the following tables refer to CPUs, full use of a logical CPU (core) is assumed. For example, if eight CPUs are required, then you may provide them in the following ways:

- Eight virtual CPUs in your virtualization platform, such as VMware Infrastructure.
- Four dual core physical CPUs.
- Two quad core physical CPUs.

Appliance sizing guidelines

<table>
<thead>
<tr>
<th>Resource</th>
<th>POC</th>
<th>Baseline</th>
<th>Datacentre</th>
<th>Consolidated Enterprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPUs</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4 to 8</td>
</tr>
<tr>
<td>RAM (GB)</td>
<td>2 to 4 (see note below)</td>
<td>4 to 8</td>
<td>8 to 16</td>
<td>16 to 32 or more</td>
</tr>
<tr>
<td>Swap Space (GB)</td>
<td>4 to 8</td>
<td>8 to 16</td>
<td>16 to 32</td>
<td>16 to 32</td>
</tr>
<tr>
<td>DB Disk (GB) - No backup</td>
<td>37</td>
<td>100</td>
<td>200</td>
<td>200 to 660</td>
</tr>
<tr>
<td>DB Disk (GB) - With local backup</td>
<td>37</td>
<td>200</td>
<td>400</td>
<td>450 to 1300</td>
</tr>
</tbody>
</table>

The disk configuration utility uses the following calculation to determine the best swap size.
• Where the amount of memory is less than 16 GB swap size is set at double the memory size.
• Where the amount of memory is between 16 and 32 GB swap is set at 32 GB.
• Where the amount of memory exceeds 32 GB swap is set to equal them memory.

The disk requirement with local backup is lower than in previous versions as the appliance backup feature does not use as much disk space as appliance snapshot when creating the backup.

⚠️ Memory requirements for POC class

2GB RAM is sufficient for normal operation, but is insufficient to activate a new TKU. To activate a TKU requires 4GB of RAM. You can increase the memory for activation and then reduce it for normal operation if required.

Performance benchmarks and tuning

The following topics contain information about the performance benchmarks and tuning.

• Disk IO Performance Guidelines
• Factors affecting performance
• Network usage
• Performance data

Disk IO Performance Guidelines

BMC publishes Performance data in order to provide the BMC Atrium Discovery user information that is meant to help determine if a particular system used to host BMC Atrium Discovery is powerful enough to discover the estate at the desired frequency.

The performance of a BMC Atrium Discovery appliance (whether Physical or Virtual) is dependent on a number of characteristics (such as number of CPUs and CPU speed), but the performance profile of the I/O subsystem is particularly important and is harder to quantify. The metrics published on this page are all gathered on systems that are based on the recommended HP servers that run a disk controller with a battery-backed write cache and are the basis for any BMC Atrium Discovery data provided.

⚠️ A BMC Atrium Discovery appliance where the I/O subsystem does not match or exceed the performance shown below will not achieve the same levels of discovery, reporting and consolidation performance.
IoZone

To provide guidance on the necessary I/O performance, performance profiles have been produced based on the powerful disk I/O performance measurement tool called IoZone. IoZone is pre-installed on the BMC Atrium Discovery appliance.

This tool can be configured to run various customized I/O tests, and the following benchmark data is produced using IoZone test that mimics the disk access profile of the BMC Atrium Discovery product as closely as possible.

To achieve this, the following command is run on a number of files:

```
/opt/iozone/bin/iozone -a -i 0 -i 2 -g 32G -q 32k -y 32k -f <filename>
```

where `<filename>` is a file on the disk being profiled. The tideway services should be stopped during the test. For more information about the options used, see the IoZone Documentation (PDF).

**NOTE:** You should stop the ADDM services prior to any iozone tests so that they are not being impaired by parallel disk activity.

This results in the execution of four tests:

- Consecutive Write
- Consecutive Read
- Random Write
- Random Read

The two *random* tests are the ones that reflect the BMC Atrium Discovery IO access profile; the other two tests are not used but cannot be disabled. Each test is performed against a range of files with sizes ranging from 64Kb to 32Gb.

The results of running this test against a HP DL 385 G5 are shown in the following graph. Click on it to see the full version:

The data behind the graph is available in raw form here:

<table>
<thead>
<tr>
<th>File Size KB</th>
<th>Random Read Kb/sec</th>
<th>Random Write Kb/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>256003</td>
<td>901408</td>
</tr>
<tr>
<td>File Size KB</td>
<td>Random Read Kb/sec</td>
<td>Random Write Kb/sec</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>128</td>
<td>4266666</td>
<td>1777775</td>
</tr>
<tr>
<td>256</td>
<td>4196720</td>
<td>1753423</td>
</tr>
<tr>
<td>512</td>
<td>3820896</td>
<td>1673203</td>
</tr>
<tr>
<td>1024</td>
<td>2348624</td>
<td>1189314</td>
</tr>
<tr>
<td>2048</td>
<td>2096212</td>
<td>1050256</td>
</tr>
<tr>
<td>4096</td>
<td>2160049</td>
<td>1055670</td>
</tr>
<tr>
<td>8192</td>
<td>2207491</td>
<td>1027338</td>
</tr>
<tr>
<td>16384</td>
<td>2181915</td>
<td>1023744</td>
</tr>
<tr>
<td>32768</td>
<td>2165047</td>
<td>949411</td>
</tr>
<tr>
<td>65536</td>
<td>2174602</td>
<td>1007874</td>
</tr>
<tr>
<td>131072</td>
<td>2170820</td>
<td>962759</td>
</tr>
<tr>
<td>262144</td>
<td>1557488</td>
<td>795433</td>
</tr>
<tr>
<td>524288</td>
<td>2117103</td>
<td>946086</td>
</tr>
<tr>
<td>1048576</td>
<td>2165387</td>
<td>977598</td>
</tr>
<tr>
<td>2097152</td>
<td>1532333</td>
<td>796689</td>
</tr>
<tr>
<td>4194304</td>
<td>1876090</td>
<td>837364</td>
</tr>
<tr>
<td>8388608</td>
<td>1895734</td>
<td>81144</td>
</tr>
<tr>
<td>16777216</td>
<td>17200</td>
<td>21388</td>
</tr>
<tr>
<td>33554432</td>
<td>6382</td>
<td>18270</td>
</tr>
</tbody>
</table>

Factors affecting performance

The following lists present a non-exhaustive catalog of factors which might affect the performance of BMC Atrium Discovery. The first list includes factors which can be configured to help improve performance. This is followed by a list of non-configurable factors.

Configurable performance factors

- **Automatic grouping** — Automatic grouping is enabled by default. Should the information provided by this feature not be required then it can be disabled. Scanning performance has been seen to improve by up to 10% when this feature is disabled.
- **CMDB synchronization** — To obtain the maximum synchronization performance when using CMDB synchronization with BMC Atrium Discovery, you should consider tuning the database that BMC Atrium CMDB (or BMC Remedy AR System) uses. See the “BMC Remedy AR System Server 7.6 Performance Tuning for Business Service Management” white paper for more information.
  The following sections in the white paper are particularly relevant:
  - Tuning an Oracle server (page 37).
- Best practices for tuning Oracle database servers (page 44).
- Tuning a SQL Server database (page 47).
- Best practices for tuning SQL Server database servers (page 49).
- **Concurrent discovery requests** — The number of concurrent Discovery requests might impact performance in environments where the network is particularly slow to respond. See Configuring discovery settings for information about how to change this.
- **Consolidation** — If the appliance is configured as a scanning appliance and is sending data to one or more consolidation appliances then you can expect a performance drop of between 5 and 10%.
- **Database cache size** — The datastore makes use of an in-memory cache which is used for all read and write operations. If the datastore is spending a lot of time swapping pages in and out of the cache, it can cause performance problems.
- **DDD removal** — If discovery (or consolidation) is in progress for most of the available time, contention might exist between the removal (aging out) of DDD nodes and the creation of new nodes. This can affect the performance of in-progress discovery runs. You can avoid this performance impact by scheduling DDD removal blackout windows during which no DDD removal is undertaken.
- **IP optimization** — Reducing the Scan Optimization Timeout will cause a degradation of appliance performance. This is because where a host has more than one IP address, a full Discovery is performed for each of those IP addresses.
- **Log level** — By default this is INFO. A log level of DEBUG will cause a degradation of appliance performance.
- **Overlapping AD Windows proxy IPs** — Where multiple Windows proxies scan the same range of IP addresses, an attempt will be made to login by each Windows proxy if no successful login has been achieved by any Windows proxy.
- **Number of CPUs** — The number of ECA engines will scale appropriately to the number of CPUs. In turn, this will increase the throughput of events through the engine. The maximum number of ECA Engines that will be automatically provisioned is 6. This setting will be chosen on a system that has 8 or more CPUs. Increasing the ECA Engine count above 6 has not been shown to further increase scanning performance.
- **Number of credentials** — For each credential whose key matches the IP address, an attempt will be made to login if no successful login has been achieved.
- **Types of patterns** — Some patterns are more complex than others and take longer to run. Therefore, if your network includes a significant number of these complex patterns, it might increase the overall run time. Care should be taken when building custom patterns and applying them to BMC Atrium Discovery. If you have any questions on this matter, contact Customer Support.

**Non-configurable performance factors**

- **Host complexity** — Discovered hosts which are information-rich, and therefore more complex than those which contain less data, might also cause a degradation of appliance performance, particularly when a large contiguous block of hosts are scanned.
• **Responses from switches for non-routable IP addresses** — Where any response is received for a scanned IP address, BMC Atrium Discovery assumes that there is an IP device at that address. If a switch is configured to send a response for a non-routable, or unassigned IP address, this might cause a degradation of appliance performance.

• **Scan ordering** — Where response time (including network latency) is an issue, this is not noticeable for individual isolated hosts. However, a contiguous block of slowly responding hosts or a high latency network segment can reduce performance noticeably. If all addresses have slow response times, then any performance degradation would be more noticeable.

**Virtual appliance performance**

When deploying a BMC Atrium Discovery Virtual Appliance into your virtualisation infrastructure the following should also be considered.

• **Other virtual machines** — If the VMware ESX instance that is hosting the BMC Atrium Discovery Virtual Appliance hosts other Virtual Machines that place a significant load on the ESX system, the performance the BMC Atrium Discovery might be affected.

• **Virtual hard disk performance** — When actively scanning, BMC Atrium Discovery places a heavy load on the database both in terms of read as well as write access. Consequently the performance of BMC Atrium Discovery will be impacted if the physical systems upon which the virtual disks are provisioned have slow response times. For example: BMC Atrium Discovery should perform better if the VMware ESX datastore on which BMC Atrium Discovery database resides maps to a physical disk that is local to the ESX instance, than if the ESX datastore maps to a remote Network Attached Storage. See **Disk IO Performance Guidelines** for more detail on the I/O requirements of the system.

**Network usage**

BMC Atrium Discovery, through its interaction with the IT infrastructure, generates some network traffic. Based on actual BMC Atrium Discovery deployments, a typical peak load of about 3 Megabits per second can be expected on the appliance. However, it is likely that differences between environments will result in peaks of network load larger than that. Some factors that could influence variations in peak load include:

• **Custom patterns** — Most of the network load is generated by BMC Atrium Discovery scanning the target environment. In addition to the core discovery required to build Host nodes, discovery requests resulting from patterns will generate some network traffic. BMC builds the patterns that are shipped in the Technology Knowledge Updates (TKU) to limit the amount of data that is requested. Similar care should be taken when developing custom patterns for your environment. For example, if a pattern were to retrieve the contents of a very large file that is common in the target environment, this could result in a larger consumption of network bandwidth.

• **Consolidation** — Consolidation results in the transfer of data between appliances. Consequently, some additional network load will be added.
• **Moving appliance backups** — It is possible to move appliance backups to or from BMC Atrium Discovery appliances; however, higher network traffic will occur during the transfer of these potentially large files.

• **Clusters** — With Cluster deployment, BMC Atrium Discovery might consume more bandwidth than a standalone setup for the following reasons:
  - **Management overhead** — The coordinator bandwidth for group coordination and additional traffic between cluster members is negligible in a normal situation. However while the cluster is recovering from failure of one or more members, the bandwidth consumption increases significantly.
  - **Concurrent discovery** — As the cluster size grows, the bandwidth consumed by the cluster members increases with the rate of discovery, as the cluster members share the discovery workload.

For better performance, you should co-locate the cluster members on a switched network with low latency.

**Performance data**

Performance figures are guidelines only and will differ between environments. Based on observations in real customer deployments it has been observed that it is realistic to be able to scan around 500 Hosts per hour. There are certain factors, configurable and otherwise that can affect performance. These are described in the Factors affecting performance section.

When scanning with BMC Atrium Discovery, the rate at which the IPs can be processed will vary depending on how many of them are actually devices, as opposed to empty space. For example, scanning a range of 100 IPs, where only one is in use, will behave differently to one where there are ninety in use.

**Consolidation**

As described in Consolidation a BMC Atrium Discovery instance can be configured as a Consolidation Appliance that receives data from multiple Scanning Appliances.

A Consolidation Appliance can process scan data at the same rate as an equivalent Scanning Appliance can. Therefore the time required to consolidate the data from all configured Scanning Appliances is approximately the same were you to sum the time taken by all Scanning Appliances to perform their scans.

For example if you had five Scanning Appliances each taking one hour to scan, then the corresponding Consolidation Appliance might take 5 hours to consolidate all this data.

This should be taken into account when planning your deployment as it is possible to end up in a situation where the Consolidation cannot keep up with the amount of data being sent to it by the Scanning Appliances.
System requirements

This section contains information on the hardware and virtualization requirements for BMC Atrium Discovery. It contains the following sections:

- Appliance specification
- Windows proxy deployment
- Windows proxy compatibility matrix
- Supported browsers

Appliance specification

The appliance specification describes the platforms used for the BMC Atrium Discovery appliance. It provides a brief introduction to the features of the appliance, and the basic tasks that need to be performed to start the appliance for the first time on a customer site. It also provides specifications for each of the appliance models.

The Appliance Specification contains the following sections:

- Appliance hardware platforms
- Supported virtualization platforms

This section is intended as a brief overview and is not comprehensive. It contains information taken from the documentation supplied with the appliance hardware. You are strongly advised to read the documentation to fully familiarize yourself with the appliance hardware and its operation.

Appliance hardware platforms

BMC Atrium Discovery is no longer supplied on physical platforms, it is supplied as a virtual appliance which is described in the Virtual Appliance section, or as a kickstart DVD which can be installed onto your own hardware.

The BMC Atrium Discovery appliance platforms are x86-64 based hardware that is supported by Red Hat Enterprise Linux 6 (64-bit only). Only servers can be used in a production environment. Workstations and laptops are not up to production specification.

See Hardware requirements for more information.

Supported virtualization platforms

The BMC Atrium Discovery Virtual Appliance is supplied as an OVF (Open Virtualization Format) file and is only supported on the VMware platforms described in this section.

For production use:

- VMware Virtual Infrastructure (ESX/ESXi) --- 4.1 and later
For testing and development

- VMware Workstation --- 8.0.2 and later
- VMware Player --- 5.0.3 and later

Windows proxy deployment

For detailed discovery of the Windows part of the target environment, BMC Atrium Discovery uses a Windows proxy, a discovery proxy that can be installed on any suitable Windows computer in the target environment.

There are two Windows proxy types available, and are selected depending on whether credential, workgroup, or active directory security is used. The preferred approach is to use the active directory proxy, installed as a service that is set to run as the discovery user. In this manner, a Windows administrator can set the user name and password, and it never has to be known or entered by a BMC Atrium Discovery administrator. In the case of the credential Windows proxy, user names and passwords for discovery are entered directly into the appliance UI.

The Windows proxy can be a physical or virtual server, and it can use an existing infrastructure server.

The following table provides information about the compatibility between Windows proxy types and versions, and the operating systems that the Windows proxy runs on for BMC Atrium Discovery.

<table>
<thead>
<tr>
<th>Windows Proxy Type</th>
<th>Earliest Compatible Windows Proxy Version</th>
<th>Windows Proxy Available for Supported Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Windows proxy</td>
<td>8.2 — with update to default SSL keys. 9.0 — no actions required.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only Windows 2008 - Service Pack 2 (x86 and x86_64) Windows 2008 R2 Windows 2012 Windows 2012 R2</td>
</tr>
<tr>
<td>Active Directory Windows proxy</td>
<td>8.2 — with update to default SSL keys. 9.0 — no actions required.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only Windows 2008 - Service Pack 2 (x86 and x86_64) Windows 2008 R2 Windows 2012 Windows 2012 R2</td>
</tr>
</tbody>
</table>

⚠️ Workgroup Windows proxy

The Workgroup Windows proxy is no longer supported. Running the Active Directory Windows proxy under a Workgroup account provides exactly the same functionality as the old Workgroup Windows proxy.
Minimum host specification

The following are the minimum recommended specifications for the Windows proxy host:

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>As stated in tables above</td>
</tr>
<tr>
<td>CPU</td>
<td>2GHz Intel Pentium® 4 CPU 512k Cache (or equivalent from other manufacturer)</td>
</tr>
<tr>
<td>Memory</td>
<td>2GB</td>
</tr>
<tr>
<td>Hard disk</td>
<td>60GB</td>
</tr>
</tbody>
</table>

To avoid any impact during resource-intensive periods of discovery, it is strongly recommended not to install the Windows proxy on any host supporting other business services. This is true even if the minimum Windows proxy specification is exceeded, since the Windows proxy will attempt to use what resources are available, in order to optimize scan throughput.

Windows discovery communications

You should also consider the ports that will need to be opened in any firewall between the appliance and the proxy or proxies, and the proxies and target hosts.

Windows discovery metadata

Discovery metadata covers Windows as well as UNIX. This provides information about why sessions failed to be established and why scripts failed to run, including information about what credential or Windows proxy was used.

Windows proxy compatibility matrix

The following table provides information about the compatibility between Windows proxy types and versions, and the operating systems that the Windows proxy runs on for BMC Atrium Discovery.

<table>
<thead>
<tr>
<th>Windows Proxy Type</th>
<th>Earliest Compatible Windows Proxy Version</th>
<th>Windows Proxy Available for Supported Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Windows proxy</td>
<td>8.2 — with update to default SSL keys.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only</td>
</tr>
<tr>
<td></td>
<td>9.0 — no actions required.</td>
<td>Windows 2008 - Service Pack 2 (x86 and x86_64)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2008 R2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012 R2</td>
</tr>
<tr>
<td>Active Directory Windows proxy</td>
<td>8.2 — with update to default SSL keys.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only</td>
</tr>
<tr>
<td></td>
<td>9.0 — no actions required.</td>
<td>Windows 2008 - Service Pack 2 (x86 and x86_64)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2008 R2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012 R2</td>
</tr>
</tbody>
</table>

Workgroup Windows proxy
The Workgroup Windows proxy is no longer supported. Running the Active Directory Windows proxy under a Workgroup account provides exactly the same functionality as the old Workgroup Windows proxy.

**Minimum host specification**

The following are the minimum recommended specifications for the Windows proxy host:

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>As stated in tables above</td>
</tr>
<tr>
<td>CPU</td>
<td>2GHz Intel Pentium® 4 CPU 512k Cache (or equivalent from other manufacturer)</td>
</tr>
<tr>
<td>Memory</td>
<td>2GB</td>
</tr>
<tr>
<td>Hard disk</td>
<td>60GB</td>
</tr>
</tbody>
</table>

To avoid any impact during resource-intensive periods of discovery, it is strongly recommended not to install the Windows proxy on any host supporting other business services. This is true even if the minimum Windows proxy specification is exceeded, since the Windows proxy will attempt to use what resources are available, in order to optimize scan throughput.

**Windows discovery communications**

You should also consider the ports that will need to be opened in any firewall between the appliance and the proxy or proxies, and the proxies and target hosts.

**Windows discovery metadata**

Discovery *metadata* covers Windows as well as UNIX. This provides information about why sessions failed to be established and why scripts failed to run, including information about what credential or Windows proxy was used.

**Supported browsers**

There are several levels of browser support:

**Comprehensive**

On such browsers, we have thoroughly tested the functionality and will consider it a bug if something does not work. These are the browsers we recommend using.

- Firefox 24.0 and above
- Chrome latest version
- Internet Explorer 8 (The *Software Context* view is not supported on Internet Explorer 8)
- Internet Explorer 9 (The *Software Context* view is not supported on Internet Explorer 9)
- Internet Explorer 10 (Desktop mode only, not Windows UI/Metro mode - see below)
- Internet Explorer 11 (Desktop mode only, not Windows UI/Metro mode - see below)
### Internet Explorer Compatibility Mode

Internet Explorer Compatibility Mode is not supported, as this forces the browser to behave as an older (unsupported) version. This behavior can be controlled with the Internet Explorer settings:

- **Tools > Compatibility View**
- **Tools > Compatibility View Settings**

### Ad-hoc

On such browsers, no formal testing has been undertaken, although some informal testing has happened and the product might work.

- Opera 11
- Safari 4
- Internet Explorer 10 Windows UI/Metro mode (Applies to Windows 8 and 8.1 only. The steps shown for Internet Explorer 11 below may also be necessary)
- Internet Explorer 11 Windows UI/Metro mode (Applies to Windows 8 and 8.1 only. Address server certificate errors in the Desktop version of the browser before attempting to use the browser in Metro mode. Also while in the desktop version of the browser: Select Internet Options, click the Security tab, select Local Intranet, then click Sites. Click Advanced in the popup that appears, and add the appliance address.)

### Deprecated

The following browsers are no longer supported by BMC Atrium Discovery. If you access the BMC Atrium Discovery user interface with one of these browsers, a warning message is displayed. You can continue to use the browser, though some features might not work.

- Internet Explorer 6
- Internet Explorer 7

Support for Compatibility and Quirks modes for the versions listed is also deprecated in later Internet Explorer browsers.

### Remote access

We test the major browsers’ latest versions at release time on the major Windows desktops and on a selection of Linux systems. Occasionally these browsers do not work as well using remote access such as RDP, VNC and virtual desktop infrastructure systems. If you encounter such a problem we are happy to try and help but cannot guarantee being able to work around all browser remote access problems.
Known issues

For a list of known issues for browsers that do not have comprehensive support, see Limitations and restrictions of this version.

Firefox internal PDF viewer

The Firefox internal PDF viewer does not display the PDF reports correctly. To work around this problem, use an alternative plugin, or download the files and view in a standalone PDF viewer. See the Mozilla website for more information. Affected versions include Firefox 21.0.

Licensing entitlement

BMC Atrium Discovery does not use a license management system. Our licensing model is simple, the customer pays an amount for a license to model up to a set number of OSIs (Operating System Instances), effectively this means hosts. A customer is entitled and licensed based on their agreement with BMC and a licensing compliance team at BMC helps customers ensure that their usage of BMC Atrium Discovery stays in line with their agreement.

Storage discovery is a separately licensed add on. It consists of regularly updated downloadable patterns in a similar way to an extended data pack or TKU. For information on purchasing licenses for storage discovery, contact your BMC Account Manager.
Installing

This section provides information on installing BMC Atrium Discovery. It contains the following sections:

- Installation, migration, and upgrade overview
- Third party software support
- Performing installation from the Kickstart DVD
- Installing the virtual appliance
- Configuring the Virtual Appliance
- Virtual Appliance
- Configuring after installation

Installation, migration, and upgrade overview

This section describes the installation, migration, and upgrade options that are available from previous versions of BMC Atrium Discovery.

Upgrade

You can only upgrade to BMC Atrium Discovery version 10.2 from version 10.1 This is described in Upgrading to version 10.2.

No architecture changes required

No architecture changes required

When upgrading from 9.x to 10.1 and from there to 10.2 there is no need to change your deployment architecture. Rather, BMC strongly recommend that you upgrade using your existing architecture and test version 10.x in a known configuration. If testing reveals that you require the performance improvements offered by a cluster, you can then add hardware and replace one or more version 10.x machines with a cluster.

Installation

New installations of BMC Atrium Discovery version 10.2 are installed from a kickstart DVD or a downloaded Open Virtualization Format (OVF) virtual machine. See Performing installation from the Kickstart DVD and Installing the virtual appliance for more information.

Upgrading from BMC Atrium Discovery version 10.1

BMC Atrium Discovery version 10 introduced a UI based upgrade which enables you to upgrade standalone machines and clusters from a single UI.
Third party software support

BMC Atrium Discovery has always been supplied as an appliance based solution. This enables the configuration of the solution to be tightly controlled, which in turn provides better supportability and lower costs, while maintaining a configuration optimized for the particular requirements of the BMC Atrium Discovery application. Another benefit of the appliance based approach is security which is described in detail here.

The benefits of the tightly controlled appliance can be broken easily by installing unsupported third party software. Of course there may be situations in which your organization's security policies may require certain software be installed. For example, anti-virus, monitoring, and authentication software. These must be restricted to a minimum. There have been cases where installing third party software has caused issues with BMC Atrium Discovery. To resolve issues on appliances where third party anti-virus and intrusion detection software has been installed, BMC may request that the third party software be removed to determine if it is the root cause of the issue.

The way in which BMC Atrium Discovery is licensed provides the customer with a license to use the Red Hat Enterprise Linux (RHEL) OS for BMC Atrium Discovery alone. It is not intended to be, or licensed as, a general purpose Linux server.

Supported software

There are only two software products that have been certified for installation on the appliance, and their use is supported.

- BMC PATROL
- BMC Atrium Orchestrator Lightweight Activity Peer

Risks involved in using unsupported third party software

If it is essential to install unsupported third party software on the appliance, you must carefully consider the risks and implications against the benefits of doing so.

Known incompatibilities

The following categories of software or operations must be avoided in all cases:

Patch Management Software

Patch management software may introduce conflicts with the installed packages comprising the customized RHEL6 operating system that the appliance uses. An operating system update mechanism is provided as a part of your BMC Atrium Discovery license entitlement. Monthly updates, where required, are available to be downloaded from the BMC EPD site. This is the only supported way of updating the operating system.
Backup Software

Extreme caution needs to be paid if the third party software is a backup solution. The BMC Atrium Discovery datastore does not support hot backups. Before backing up, all BMC Atrium Discovery services must be shut down and restarted once the backup is complete. If you do not do so and then restore the backup, then the restored datastore will be unrecoverably corrupted.

A simpler and more reliable option is to use the BMC Atrium Discovery backup and restore function.

Installation of software using RPM packages

You must avoid installing software using RPM commands. In addition to installing the third party software, these are very likely to update system libraries which may be incompatible with those currently used, or when the operating system or the BMC Atrium Discovery application are updated later.

Performing installation from the Kickstart DVD

BMC Atrium Discovery is supplied as a virtual appliance and a kickstart DVD image. To install a virtual appliance, see Installing the virtual appliance.

Installing from the kickstart DVD

To install BMC Atrium Discovery from a kickstart DVD, first download the kickstart DVD image from the BMC Electronic Product Distribution (EPD) site. The image is large, so when writing the image to a physical DVD, ensure that you use a dual layer DVD and burner.

The kickstart DVD can be used to install BMC Atrium Discovery on certain customer supplied hardware. See prerequisites.

- Third party software

BMC Atrium Discovery is built as an appliance that is not intended to have any additional software installed on it. If you have an urgent business need to install additional software on the appliance, please read the third party software support page.

- No OS customizations are supported on the appliance

The BMC Atrium Discovery software is delivered as an appliance model (either virtual or physical), and includes the entire software stack from a Linux OS to the BMC application software. The OS must not be treated for general purpose use, but rather as a tightly integrated part of the BMC Atrium Discovery solution. As such, customizations to the OS should only be made at the command line level if explicitly described in this online
If you have an urgent business need to change any OS configuration, you must contact BMC Customer Support before doing so.

**Unsupported options**

As part of the OS installation you are presented with options that are not supported as part of BMC Atrium Discovery. For example:

- Encrypt disks option in the partitioning screens
- Advanced Storage options.

These are not supported.

**Prerequisites**

Before installing BMC Atrium Discovery from the kickstart DVD, you should ensure that you satisfy the following prerequisites:

- You must be an experienced Linux system administrator.
- You must have read the Hardware requirements page for clusters, and for standalone appliances, the sizing guidelines documentation for the current version of the product, included Appliance sizing guidelines below.
- You must be installing on x86-64 based hardware that is supported by Red Hat Enterprise Linux 6 (64-bit only).
- If you intend to run BMC Atrium Discovery in FIPS compliant mode, you must read Running in FIPS compliant mode.

A facility for installation on a 32 bit machine is not available.

**Appliance sizing guidelines**

There are many factors to be taken into consideration when specifying the configuration of the appliance. Every environment is different and consequently the data published here is purely a guide as to how to configure your appliance.

There are many factors to be taken into consideration when specifying the configuration of the appliance. Every environment is different and consequently the data published here is purely a guide as to how to configure your appliance.

The following initial guidelines are based on typical deployments of standalone systems in the field, and are intended to serve only as recommended configurations for your environment.
This section defines four "classes" of appliance deployment that broadly follow how BMC Atrium Discovery is deployed in the field. They are differentiated by how many Operating System Instances (OSIs) that are being scanned by BMC Atrium Discovery. The names given to these classes are of use only in this document and do not relate to the various editions that BMC Atrium Discovery is available in.

The classes are:

- **Proof of Concept.** Small, time-limited test deployments of BMC Atrium Discovery, scanning up to 150 OSIs.
- **Baseline.** A typical baseline as offered by BMC. Scanning up to 500 OSIs.
- **Datacentre.** A typical large scale deployment. Scanning up to 5000 OSIs.
- **Consolidated Enterprise.** Enterprise scale deployments, typically a Consolidation Appliance taking feeds from many Scanning Appliances. Typically scanning or consolidating of the order of 20000 OSIs or more. At these levels, a weekly scanning or focused scanning strategy may need to be adopted.

### Proof of Concept

The Proof of Concept class has minimal storage allowance as they are only intended for a limited period of scanning such as a week long trial. For longer periods or a continuously used development or UAT system, the Baseline class is the minimum recommended.

### Memory and swap considerations

The recommended figures for memory provide a good level of performance in typical scenarios. The upper level should not be considered a limit, as BMC Atrium Discovery will make use of all available memory. You can determine whether additional memory is needed in your appliance by monitoring swap usage.

The recommended figures for swap can be exceeded; there is no harm in doing so, and having more swap can mean more RAM is available for buffers and caches, improving performance. All virtual appliances are initially configured with 8 GB swap.

### Memory and swap usage

Memory and swap usage depends on the nature of the discovery being performed, with Mainframe and VMware (vCenter/vSphere) devices requiring more than basic UNIX and Windows devices.
Where the following tables refer to CPUs, full use of a logical CPU (core) is assumed. For example, if eight CPUs are required, then you may provide them in the following ways:

- Eight virtual CPUs in your virtualization platform, such as VMware Infrastructure.
- Four dual core physical CPUs.
- Two quad core physical CPUs.

### Appliance sizing guidelines

<table>
<thead>
<tr>
<th>Resource</th>
<th>POC</th>
<th>Baseline</th>
<th>Datacentre</th>
<th>Consolidated Enterprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPUs</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4 to 8</td>
</tr>
<tr>
<td>RAM (GB)</td>
<td>2 to 4 (see note below)</td>
<td>4 to 8</td>
<td>8 to 16</td>
<td>16 to 32 or more</td>
</tr>
<tr>
<td>Swap Space (GB)</td>
<td>4 to 8</td>
<td>8 to 16</td>
<td>16 to 32</td>
<td>16 to 32</td>
</tr>
<tr>
<td>DB Disk (GB) - No backup</td>
<td>37</td>
<td>100</td>
<td>200</td>
<td>200 to 660</td>
</tr>
<tr>
<td>DB Disk (GB) - With local backup</td>
<td>37</td>
<td>200</td>
<td>400</td>
<td>450 to 1300</td>
</tr>
</tbody>
</table>

The disk configuration utility uses the following calculation to determine the best swap size.

- Where the amount of memory is less than 16 GB swap size is set at double the memory size.
- Where the amount of memory is between 16 and 32 GB swap is set at 32 GB.
- Where the amount of memory exceeds 32 GB swap is set to equal them memory.

The disk requirement with local backup is lower than in previous versions as the appliance backup feature does not use as much disk space as appliance snapshot when creating the backup.

⚠️ **Memory requirements for POC class**

2GB RAM is sufficient for normal operation, but is insufficient to activate a new TKU. To activate a TKU requires 4GB of RAM. You can increase the memory for activation and then reduce it for normal operation if required.

BMC recommend that for the best performance, you use two logical disks. For single disk installations your sizing calculations should be based on the size of the database (see the table above) plus the size of the OS disk (146 GB).

### Installing BMC Atrium Discovery

⚠️ **Partitioning destroys all data on disks**
Installing BMC Atrium Discovery involves partitioning your disks. Partitioning disks destroys any data on those disks. You should understand partitioning before installing BMC Atrium Discovery.

To install BMC Atrium Discovery from a kickstart DVD:

1. Boot your host using the kickstart DVD. See the documentation supplied with the hardware platform for information on this. You are presented with a splash screen which enables you to select installer options (press F2 to see more information). The options are:
   - `install` to install on a system configured as specified in Appliance hardware platforms. This option performs an installation that completely overwrites any data on the system, and enables you to set network configuration, keyboard layout, language and timezone. BMC Atrium Discovery will always be installed onto the first disk. Additional disks can be configured from the BMC Atrium Discovery UI when the install is complete.
   - `custom` This option enables you to customize disk partitioning, but is otherwise identical to the `install` option. Always use the `install` option in preference to the `custom` option, even if you have multiple disks. Should you use `custom` to create a non-standard disk layout, the Disk Configuration feature will not be able to manage BMC Atrium Discovery storage. When using the `custom` option, ensure that the /boot partition is 250 MB as in the default partitioning scheme.

2. Enter one of the supported options at the `boot: prompt and press enter. The Red Hat Enterprise Linux installer starts.
3. If you have more than one network interface, you are prompted to choose one. Select the first ethernet device, eth0, and click OK.
4. When the TCP/IP configuration screen is displayed, enter the required networking information and click OK. If you select manual configuration, one or more further configuration screens will be presented. Provide the requested information.
5. If the disks are not recognized or have never been partitioned, a partitioning table warning is displayed. Click Yes to proceed.
6. When prompted for the language to use during installation, select English.
7. When prompted, choose the appropriate keyboard for the system.
8. When the time zone configuration screen appears:
9. If you selected the custom option, use the following steps to configure the disks.

**Non standard disk configuration**

These custom install instructions will result in a valid disk layout identical to that created by the install option. The disk configuration utility cannot be able to manage disks if volume groups or a non standard disk configuration are used.

a. Select the Replace Existing Linux System(s) installation type.
b. Do not select the Encrypt system option.
c. Select the Review and modify partitioning layout option.
d. Click next. On the storage device selection screen, select the first disk, and click the right arrow to move it into the Install Target Devices list.
e. Click next. The Review Partitioning screen is displayed.

f. Delete the volume group shown beneath LVM Volume Groups or create the partitions in this volume using the sizes described in the table below as a guide. If you do not delete the LVM volume group, you will not be able to use the disk configuration utility to manage disks.

g. If you have deleted the LVM volume group, delete all remaining partitions on all disks and enter the following partition information for the first disk:

<table>
<thead>
<tr>
<th>Mountpoint</th>
<th>Type</th>
<th>Size (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>swap</td>
<td>ext4</td>
<td>16384</td>
</tr>
<tr>
<td>/boot</td>
<td>ext4</td>
<td>250</td>
</tr>
<tr>
<td>/var</td>
<td>ext4</td>
<td>1768</td>
</tr>
<tr>
<td>/var/log</td>
<td>ext4</td>
<td>896</td>
</tr>
<tr>
<td>/var/log/audit</td>
<td>ext4</td>
<td>896</td>
</tr>
<tr>
<td>/tmp</td>
<td>ext4</td>
<td>2048</td>
</tr>
<tr>
<td>/home</td>
<td>ext4</td>
<td>1536</td>
</tr>
<tr>
<td>/</td>
<td>ext4</td>
<td>1024</td>
</tr>
<tr>
<td>/usr</td>
<td>ext4</td>
<td>100 (Fill to maximum allowable space.)</td>
</tr>
</tbody>
</table>

See Appliance sizing guidelines for information on recommended swap size.
h. If you have more than one disk, do not configure them at this time. Instead use the disk configuration utility when you have installed BMC Atrium Discovery.

10. The partitioning and installation of the OS and BMC Atrium Discovery begins. This might take some time.

11. When the installation has completed, remove the DVD and click the Reboot button. The BMC Atrium Discovery banner displays networking information.

12. The installation is now complete.

13. If you have additional disks, use the disk configuration utility to configure them.

Installing the virtual appliance

Before you begin

Before you start the installation procedure, ensure the following:

- You have identified a server onto which you will deploy your BMC Atrium Discovery virtual appliance
- The server has sufficient spare resources to support the intended configuration of the virtual appliance.
- You have identified the VMware Administrator who will be installing the virtual appliance.

BMC Software no longer provide 32 bit virtual appliances or kickstarts of BMC Atrium Discovery.

**Third party software**

BMC Atrium Discovery is built as an appliance that is not intended to have any additional software installed on it. If you have an urgent business need to install additional software on the appliance, please read the third party software support page.

**No OS customizations are supported on the appliance**

The BMC Atrium Discovery software is delivered as an appliance model (either virtual or physical), and includes the entire software stack from a Linux OS to the BMC application software. The OS must not be treated for general purpose use, but rather as a tightly integrated part of the BMC Atrium Discovery solution. As such, customizations to the OS should only be made at the command line level if explicitly described in this online documentation, or under the guidance of BMC Customer Support. If you have an urgent business need to change any OS configuration, you must contact BMC Customer Support before doing so.
Supported virtualization platforms

The BMC Atrium Discovery Virtual Appliance is supplied as an OVF (Open Virtualization Format) file and is only supported on the VMware platforms described in this section.

For production use:

- VMware Virtual Infrastructure (ESX/ESXi) --- 4.1 and later

For testing and development

- VMware Workstation --- 8.0.2 and later
- VMware Player --- 5.0.3 and later

**VMware Virtual Infrastructure (ESX/ESXi) --- 4.1 and later**

To install the BMC Atrium Discovery Virtual Appliance onto VMware Virtual Infrastructure (ESX/ESXi) version 4 and 5.

⚠️ You must add a CD-ROM drive to the virtual machine before you can install VMware tools. Do this before starting the virtual machine.

The major steps in this procedure are outlined below. For full information, see the VMware documentation.

1. Copy and extract the BMC Atrium Discovery Virtual Appliance zip file to the local file system.
2. In VMware Virtual Infrastructure, from the File menu, select Deploy OVF Template. The wizard guides you through the rest of the process described here.
3. Enter the file name of the BMC Atrium Discovery OVF file into the Filename dialog box, or click Browse... to locate the file. Click Next.
4. Review the details provided. Click Next.
5. Provide a name for the virtual machine. Click Next.
6. Choose datastore location for the virtual machine. Click Next.
7. Choose thin or thick disk format. The default is thick which pre-allocates all disk space. Click Next.
8. Review the configuration and click Finish to proceed. A progress bar is displayed while the virtual machine is deployed.
9. Click the Hardware tab and click Add.
10. Select DVD/CD-ROM Drive and click Next.
11. Accept the defaults for the drive. Optionally you can now add additional disk storage which can be managed with the disk configuration utility.
12. Click the Hardware tab and click Add.
13. Select Hard Disk and click Next.
14. Choose the disk capacity and other options. For information about recommended capacity, see the Hardware requirements.
15. Click Next.
16. Review the settings and click Finish.
17. Click OK to save the changes. You can now start the virtual machine.

**VMware Workstation — 8.0.2 and later**

To install the Virtual Appliance on VMware Workstation 6 or 8.

1. Copy and extract the BMC Atrium Discovery Virtual Appliance zip file to the local file system.
2. In VMware Workstation, from the File menu, select Open.
3. Enter the file name of the BMC Atrium Discovery OVF file into the Filename dialog box, or click select OVF as the file type and click Browse... to locate the file. Click Open.
4. Enter a name for the virtual machine and click Import. VMware Workstation imports the virtual machine and the virtual machine is displayed in the virtual machine library.

**VMware Player — 5.0.3 and later**

You cannot install or run an OVF virtual machine in Windows by double-clicking it. An error message "Invalid option t" is displayed. Instead, follow the procedure below.

To install the Virtual Appliance on VMware Player 5.0.3.

1. Copy and extract the BMC Atrium Discovery Virtual Appliance zip file to the local file system.
2. In VMware Player, from the File menu, select Open.
3. Enter the file name of the BMC Atrium Discovery OVF file into the Filename dialog box, or click select OVF as the file type and click Browse... to locate the file. Click Open.
4. Enter a name for the virtual machine and click Import. VMware Player imports the virtual machine.
5. When imported, the virtual machine appears in the virtual machine library.
Earlier versions

VMware Player 4.x can fail to build VMware Tools with the latest revision of Red Hat Enterprise Linux version 6. Some older versions of VMware tools do not natively support Red Hat Enterprise Linux version 6.

VMware Workstation version 7 and earlier and VMware Player version 3 and earlier do not directly support appliances generated by their current tools. However, you can convert the OVF file using the OVF converter in order to import it.

To convert, enter the following at a command prompt:

```
"c:\Program Files\VMware\VMware OVF Tool\ovftool"
ADDM_VA_64_8.90.5_123456_dev_Community.ovf
ADDM_VA_64_8.90.5_123456_dev_Community.vmx
```

When you have converted the file, you can install it using File > Open.

Post Installation

⚠️ Important Post Installation Steps

It is important to consider the following steps after installing.

- VMware Tools
- Installing VMware Tools
- Setting up NTP

VMware tools

VMware Tools is required but is not installed on the shipped BMC Atrium Discovery Virtual Appliance, as the correct version of the tools is dependent on the version of the VMware host. When you have installed the BMC Atrium Discovery Virtual Appliance, you must always install VMware Tools.

To install VMware Tools

The following procedure describes how to install VMware Tools. For full information, see the documentation available on the VMware Support site:

1. From VMware Infrastructure Client, right click the virtual machine in the left pane.
2. Select Install/Upgrade VMware Tools. This makes a virtual CD device available to the virtual machine.
3. Log in to the virtual machine command line, and change to the root user.

4. To determine the device name/drive name for the currently installed and connected CD-ROMs or DVD-ROMs, issue the following command:
   ```bash
cat /proc/sys/dev/cdrom/info | grep "drive name"
```

5. Issue the following commands (change the `XXX-nnnnnn` in the RPM or tar file name below to the version presented by your VMware product):
   a. `mkdir /mnt/cdrom`
   b. `mount -t iso9660 /dev/<device> /mnt/cdrom`
      where `<device>` is the device name reported in step 4.
   Enter one of the following commands depending on whether VMware Tools is supplied on the virtual CD as an RPM, or a tar file.
   i. If you have an rpm enter the following command:
      ```bash
      rpm -i /mnt/cdrom/VMwareTools-XXX-nnnnnn.x86_64.rpm
      umount /mnt/cdrom
      ```
   ii. If there is a tar file, enter the following commands:
      ```bash
      cd /tmp
      tar zxvf /mnt/cdrom/VMwareTools-XXX-nnnnnn.tar.gz
      umount /mnt/cdrom
      cd vmware-tools-distrib
      ./vmware-install.pl
      ```

6. Respond to the configuration questions on the screen. Press Enter to accept the default value.

If at any later point, you need to re-run the VMware tools script, the default location will be:
```
/usr/bin/vmware-install.pl
```

**Setting up NTP**

It is essential to configure ntp on your BMC Atrium Discovery Virtual Appliance. If significant clock skews occur then it can impact the functioning of the system or even prevent the BMC Atrium Discovery product from starting. VMware have documented their recommended timekeeping best practices for Linux based VMs. The KnowledgeBase article can be found [here](#).

**Configure the appropriate time zone**

It is sensible to configure the Virtual Appliance to be in the time zone appropriate to its location. See [Localizing the appliance](#) for more information about this.

**Next Steps**

Once the Virtual Appliance is installed it is a good time to configure it to have sufficient resources for its intended use. Doing so now will be easier than extending the resources once the Virtual Appliance has been in use for some time.

See [Configuring the Virtual Appliance](#) for details.
Configuring the Virtual Appliance

With any virtual appliance, there are some things to consider before deployment, particularly when you move into production.

⚠️ Dedicated VMware Resources

It is strongly recommended that the CPU and RAM resources that are allocated to the ADDM appliance are reserved, and are not shared with other VMware guest OSes. If this is not the case, performance might be inconsistent and might not achieve expectations. For more details contact your VMware administrator.

Virtual Appliance sizing guidelines

In previous releases, sizing guidelines were provided by defining classes of appliance deployment. The introduction of Big Discovery means that these classes are no longer valid for clustered deployments. They do still have some relevance for single appliance deployments, but where scale is required, just add hardware!

The appliance specification page in the UI retains the classes for standalone deployments, but not in clusters. We are reviewing BMC Atrium Discovery performance as version 10 is deployed, and might extend our recommendations based on our customers’ real experiences.

For more information on the previous recommendations, see Sizing guidelines in the version 9.0 documentation.

Virtual Appliance

The BMC Atrium Discovery Virtual Appliance (VA) uses VMware virtualization technology to provide Application Distribution Management. The following VAs are available:

- Community Edition: A free download of BMC Atrium Discovery that is limited to scan up to 60 OSIs.

BMC Atrium Discovery and Dependency Mapping is certified VMware Ready
Pursuant to the terms of VMware’s VMware Ready program. BMC Atrium Discovery is a “qualifying partner product” in that it has been approved by VMware as meeting all of the requirements of the VMware Ready program. BMC Atrium Discovery is certified in the Application Software category of the VMware Ready program and is fully supported by BMC. The product has been fully tested to ensure reliable operation under fully-loaded conditions. This certification can be validated via this article, in the Installing the virtual appliance guide, and on VMware’s Solution Exchange.

VMware and VMware Ready is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions.

Statement of support for VMware Virtualization Solutions

BMC Atrium Discovery is shipped as a Virtual Appliance (VA) which uses VMware virtualization technology and is fully supported on the VMware Virtualization solutions mentioned in this statement. The VAs are production-ready, tested for scalability and security, and delivered in OVF format. They are built using VMware ESX version 4.1, and are compatible with current versions of VMware ESX, ESXi, Workstation and Player. However, to deploy the Virtual Appliance into older versions, or VMware Server, an additional conversion process is required.

BMC provides Operating System updates as part of its monthly BMC Atrium Discovery updates.

Configuring after installation

This section details the configuration steps you perform after installing BMC Atrium Discovery and configuring the appliance for the first time.

- Configuring the appliance to use Tectia SSH and X.509 certificates
- Configuring a bonded NIC
- Adding static routes
- Configuring UNIX and Solaris 8 systems for discovery
- Installing BMC PATROL on a BMC Atrium Discovery system

Configuring the appliance to use Tectia SSH and X.509 certificates

The appliance can be configured to use Tectia SSH and x.509 certificates, as described in the following sections. These procedures assume that you have administrator level knowledge of Tectia SSH, and a Tectia SSH server installed on your network.

Installing and configuring the Tectia SSH client

Install the Tectia SSH client as described in the Tectia installation guide.
After you have installed and configured the client for the tideway user, you should be able to access remote servers using sshg3 from the command line, though you might need to add /opt/tectia/bin to the PATH. You should test servers that require X.509 certificates and those that do not, if possible.

**Configuring discovery to use the Tectia SSH client**

To configure Discovery to use the Tectia SSH client:

1. Create a .tideway.py file in /usr/tideway and add the following entry:

   ```
   SSH_CMD="/opt/tectia/bin/sshg3"
   ```

2. Restart the Tideway services.
   You will see deprecation warnings in the logs about use of the .tideway.py file but you can ignore them.

Discovery will now use Tectia SSH instead of OpenSSH for connections to remote systems.

**Configuring a bonded NIC**

Network Interface Card (NIC) bonding (also known as teaming) enables you to join two NICs as a single physical device so that they appear as one interface. This is usually performed to provide failover capabilities or load balancing. Bonding can be configured on the BMC Atrium Discovery hardware appliance. It is not something that you should consider on a virtual appliance, because failover and load balancing should be provided by the virtualization platform.

**Network configuration**

When changing network configuration on your appliance, you should always be able to access the system console in case the new configuration does not work correctly.

**Netadmin does not work with bonded NICs**

If you configure your appliance to use bonded NICs, you can no longer use the netadmin user to perform any networking configuration.

The BMC Atrium Discovery appliance relies on eth0 having an IP address. When the appliance starts, it requires an IP address on eth0 for configuration and start-up services. In a default bonded NIC configuration, the IP address would be assigned to the bond point bond0, and the BMC Atrium Discovery services would be unable to start without additional configuration.
The `ifconfig` output for bonded NIC cards is shown in the following example, where `bond0` has an IP address but `eth0` and `eth1` do not.

```
bond0  Link encap:Ethernet HWaddr 00:11:43:FD:9A:B1
  inet addr:192.168.0.25 Bcast:192.168.0.255 Mask:255.255.255.0
  UP BROADCAST RUNNING MASTER MULTICAST MTU:1500 Metric:1
  RX packets:58071 errors:0 dropped:0 overruns:0 frame:0
  TX packets:1465 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:4315472 (4.1 Mb) TX bytes:120360 (117.5 Kb)

eth0  Link encap:Ethernet HWaddr 00:11:43:FD:9A:B1
  UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
  RX packets:26447 errors:0 dropped:0 overruns:0 frame:0
  TX packets:1262 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:1992430 (1.9 Mb) TX bytes:95078 (92.8 Kb)
  Interrupt:16

eth1  Link encap:Ethernet HWaddr 00:11:43:FD:9A:B1
  UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
  RX packets:31624 errors:0 dropped:0 overruns:0 frame:0
  TX packets:203 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:2323042 (2.2 Mb) TX bytes:25282 (24.6 Kb)
  Interrupt:17
```

Creating a bond point called eth0

To create a bond point called `eth0` you need to:

1. Configure the network interface scripts in `/etc/sysconfig/network-scripts` to force the hardware to bond to the names `eth1` and `eth2` using their MAC addresses. Take care with the "SLAVE=yes" and "MASTER=eth0" options in each file.
2. Create the `eth0` interface which is actually assigned the IP configuration options.
3. Configure the boot process to include the `bonding` module and to use `eth0` as the bond point, with the various options associated with that bond.

To do this, log in as the `tideway` user and perform the following steps:

1. Stop the `tideway`, cluster, omniNames and appliance services. Enter:

   ```bash
   [tideway@localhost] $ sudo /sbin/service tideway stop
   [tideway@localhost] $ sudo /sbin/service cluster stop
   [tideway@localhost] $ sudo /sbin/service omniNames stop
   [tideway@localhost] $ sudo /sbin/service appliance stop
   ```
2. Change to the `/etc/sysconfig/network-scripts` directory and determine the MAC addresses (`HWaddr`) of the network cards that you want to bond. Enter:

```
[tideway@localhost] $ cd /etc/sysconfig/network-scripts
[tideway@localhost network-scripts] $ /sbin/ifconfig -a | grep HWaddr
eth0      Link encap:Ethernet  HWaddr 00:11:43:FD:9A:B1
eth1      Link encap:Ethernet  HWaddr 00:11:43:FD:9A:B2
```

In this example the MAC Addresses are as follows:
- `eth0` — `00:11:43:FD:9A:B1`
- `eth1` — `00:11:43:FD:9A:B2`

3. Change to the root user and using a text editor, edit, or create and edit `ifcfg-eth1` to contain the following:

```
DEVICE=eth1
BOOTPROTO=none
ONBOOT=yes
HWADDR=00:11:43:FD:9A:B1 # The MAC address of eth1 as determined in the previous step.
TYPE=Ethernet
MASTER=eth0
SLAVE=yes
```

4. Using a text editor, edit or create and edit `ifcfg-eth2` to contain the following:

```
DEVICE=eth2
BOOTPROTO=none
ONBOOT=yes
HWADDR=00:11:43:FD:9A:B2 # The MAC address of eth2 as determined in the previous step.
TYPE=Ethernet
MASTER=eth0
SLAVE=yes
```

5. Using a text editor, edit or create and edit `ifcfg-eth0` to contain the following. Complete the `IPADDR`, `NETMASK`, and `GATEWAY` information.

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
IPADDR=<required IP address>
```
Options such as `BROADCAST` and `NETWORK` must be defined in `ifcfg-eth0` if they are required. The bonding options, `BONDING_OPTS`, might not be required but the ones shown are good defaults. `miimon=100` means check the link state every 100 milliseconds and `mode=active-backup` provides tolerance. There are other mode options available, some of which require changes to other items.

6. Reboot the system.
7. Log in as the tideway user and run the following command:

```
[tideway@localhost ] $ /sbin/ifconfig -a
```

The output should look like the following, note the `bond0` interface:

```
bond0  Link encap:Ethernet  HWaddr 00:00:00:00:00:00
       BROADCAST MASTER MULTICAST  MTU:1500  Metric:1
       RX packets:0 errors:0 dropped:0 overruns:0 frame:0
       TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:0
       RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

eth0  Link encap:Ethernet  HWaddr 00:11:43:FD:9A:B1
       inet addr:192.168.0.5  Bcast:192.168.0.255  Mask:255.255.255.0
       inet6 addr: fe80::250:56ff:fea7:362/64 Scope:Link
       UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
       RX packets:21026 errors:0 dropped:0 overruns:0 frame:0
       TX packets:175 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:0
       RX bytes:1405048 (1.3 MiB)  TX bytes:61541 (60.0 KiB)

eth1  Link encap:Ethernet  HWaddr 00:11:43:FD:9A:B1
       UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
       RX packets:10529 errors:0 dropped:0 overruns:0 frame:0
       TX packets:87 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:1000
       RX bytes:703475 (686.9 KiB)  TX bytes:30280 (29.5 KiB)

eth2  Link encap:Ethernet  HWaddr 00:11:43:FD:9A:B1
       UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
       RX packets:10498 errors:0 dropped:0 overruns:0 frame:0
       TX packets:90 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:1000
       RX bytes:701633 (685.1 KiB)  TX bytes:31793 (31.0 KiB)
```
8. Check `/var/log/messages` to check for a successful bonding:

```
Nov 18 15:47:18 localhost kernel: Ethernet Channel Bonding Driver: v3.4.0-1 (October
Nov 18 15:47:18 localhost kernel: bonding: MII link monitoring set to 100 ms
Nov 18 15:47:18 localhost kernel: bonding: eth0 is being created...
Nov 18 15:47:18 localhost kernel: ADDRCONF(NETDEV_UP): eth0: link is not ready
Nov 18 15:47:18 localhost kernel: bonding: eth0: enslaving eth1 as an active interface
Nov 18 15:47:18 localhost kernel: e1000: eth1 NIC Link is Up 1000 Mbps Full Duplex,
Nov 18 15:47:18 localhost kernel: bonding: eth0: link status definitely up for interface
Nov 18 15:47:18 localhost kernel: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
Nov 18 15:47:18 localhost kernel: e1000: eth2 NIC Link is Up 1000 Mbps Full Duplex,
Nov 18 15:47:18 localhost kernel: bonding: eth0: enslaving eth2 as an active interface
```

The list of running kernel modules should include the `bonding` module.

9. Run the following command:

```
[tideway@localhost ] $ /sbin/lsmod | sort
ac                      38729  0
acpi_memhotplug         40517  0
asus_acpi               50917  0
ata_piix                57541  0
autofs4                 63049  3
backlight               39873  1 video
battery                 43849  0
*bonding                142537 0*
button                  40545  0
cdrom                   68713  1 ide_cd
crypto_api              42945  1 xfrm_nalgo
...
```

Adding static routes

Adding static routes to the BMC Atrium Discovery server is necessary only on rare occasions. Routing is configured on routing devices, therefore it should not be necessary to configure static routes on Red Hat Enterprise Linux servers or clients. However, if you require this type of configuration you can store the routes in a `/etc/sysconfig/network-scripts/route-interface` file. Static routes should only be configured for different subnets.

For example, static routes for the `eth0` interface would be stored in the file `/etc/sysconfig/network-scripts/route-eth0` with a format similar to the following:
ADDRESS0=172.21.22.65  
NETMASK0=255.255.255.255  
GATEWAY0=172.21.19.68  
ADDRESS1=172.21.20.0  
NETMASK1=255.255.255.0  
GATEWAY1=172.21.19.68

In this example, two static routes are added to the IP 172.21.22.65, and network 172.21.20.0/24 to the GW GATEWAY1=172.21.19.68.

For more information on configuring static routes, see the Red Hat Linux documentation. Further, you should take into consideration the warnings regarding support of OS customizations to your appliance.

**Configuring UNIX and Solaris 8 systems for discovery**

Before you begin using the BMC Atrium Discovery consider performing the following steps:

- Ensure that you have the lsof tools installed on your UNIX systems to enable adequate process to communication mapping.
- Ensure that you have the NTP service configured.
- For Solaris 8, ensure that BMC Atrium Discovery uses correct `ifconfig` binaries

**Process to Communication Mapping (LSOF)**

BMC Atrium Discovery uses the lsof command to map processes to network communications on some UNIX-like operating systems. This enables Reasoning to accurately map software applications that span multiple hosts.

Some organizations choose not to install lsof and other diagnostic tools on their UNIX systems. If lsof is not installed on your UNIX systems you will not be able to relate the communication to processes on those systems.

⚠️ **Do not use pfiles in Solaris Zones --- Potential Data Loss**

The lsof utility cannot obtain process communication information from the local zone in Oracle Solaris operating systems. Performing a search on the Internet shows possible workarounds using pfiles. Using pfiles can result in data loss on the Solaris host. Oracle highlights the danger of using pfiles in the warnings section of their own documentation.

**NTP Configuration**

You should configure the NTP service on the Virtual Machine. This is of particular importance if it will be involved in consolidation. If you do not configure NTP, the appliance time can drift backwards, preventing the datastore from validating itself. This in turn prevents the system from restarting. See Performing time synchronization for more information.
Solaris 8 and ifconfig

In Solaris 8 there are two ifconfig binaries:

- /sbin/ifconfig
- /usr/sbin/ifconfig

In all versions of Solaris other than 8, there is a single binary and a symbolic link. The default path statement set by BMC Atrium Discovery ensures that /sbin/ifconfig is called first. In Solaris 8 this is the incorrect version, /usr/sbin/ifconfig must be run to obtain the correct information. To ensure this is the case, edit the ifconfig discovery script to specify the full path to ifconfig:

```
IFCONFIG=`PRIV_IFCONFIG /usr/sbin/ifconfig -a 2>/dev/null`
echo "$IFCONFIG"
```

Do not modify the path statement to correct this issue as that will cause other problems.

Installing BMC PATROL on a BMC Atrium Discovery system

BMC Atrium Discovery is built as an appliance that is not intended to have any additional software installed on it, with the single exception of BMC PATROL.

You install BMC PATROL using the BMC Installation Utility. BMC PATROL documentation is available on the BMC documentation portal, along with a selection of PDF documentation including the Installation Utility Reference Manual. You must log in to the BMC documentation portal to view this content.

This section describes the additional steps required to modify the BMC Atrium Discovery firewall. If you are unsure of any of the steps described, contact Customer Support.

BMC PATROL requires additional ports to be open in the BMC Atrium Discovery firewall. They are:

- 50001 — required during installation only. Must be closed after installation.
- 3181 — required during operation of BMC PATROL. 3181 is the default port. If your installation of BMC PATROL uses a custom port, you should open that one instead.
- 2059 — required during operation of BMC PATROL for connection to BMC Real Time server (RTserver).
- 3183 — optional during operation of BMC PATROL for connection to BMC Proactive Performance Management (BPPM) Agent.

Any changes to the firewall configuration are reflected in the baseline status. You should rebaseline after making these changes.
Fallback firewall configuration

When BMC Atrium Discovery is installed, the default firewall is copied to create a fallback firewall.

<table>
<thead>
<tr>
<th>IP version</th>
<th>Default firewall</th>
<th>Fallback firewall</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4</td>
<td>/etc/sysconfig/iptables</td>
<td>/etc/sysconfig/iptables.fallback</td>
</tr>
<tr>
<td>IPv6</td>
<td>/etc/sysconfig/ip6tables</td>
<td>/etc/sysconfig/ip6tables.fallback</td>
</tr>
</tbody>
</table>

When the iptables or ip6tables services are restarted and the service fails because the default configurations have errors, the service will now attempt to use the appropriate fallback file instead.

To install BMC PATROL on a BMC Atrium Discovery system

This procedure provides detailed steps for modifying the BMC Atrium Discovery firewall. For the steps concerning the installation of BMC PATROL, you should consult the BMC PATROL documentation.

1. Back up iptables. As the root user, enter:

   ```
   cp /etc/sysconfig/iptables /etc/sysconfig/iptables.backup
   ```

2. Add the following line to /etc/sysconfig/iptables before the lines beginning `-A INPUT`

   ```
   -A INPUT -p tcp -m tcp --dport 50001 -j ACCEPT
   ```

3. Save the changes.
4. Restart the firewall. Enter:

   ```
   /sbin/service iptables restart
   ```

5. Download the BMC PATROL Agent web installer to BMC Atrium Discovery.
6. Run the installer using the serveronly option. Consult the BMC PATROL installation instructions for full information on the installation.
7. When you have installed BMC PATROL, close port 50001 and open port 3181 (or the custom port that your BMC PATROL installation uses) and port 2059. To do this, replace the line entered to open port 50001 with the following:

   ```
   -A INPUT -p tcp -m tcp --dport 3181 -j ACCEPT
   ```
8. If you intend to use the BPPM Agent, add the following line:

```
-A INPUT -p tcp -m tcp --dport 3183 -j ACCEPT
```

9. Save the changes.

10. Restart the firewall. Enter:

```
/sbin/service iptables restart
```
Upgrading

This section provides information on upgrading to this version of BMC Atrium Discovery. It contains the following sections:

- Upgrading BMC Atrium Discovery
- Moving node keys
- Configuring after upgrade

Upgrading BMC Atrium Discovery

You can upgrade to BMC Atrium Discovery 10.2 or later from BMC Atrium Discovery version 10.1 and later.

Upgrading from an earlier version?

If you are using an earlier version of BMC Atrium Discovery, see the appropriate upgrade documentation to upgrade to a version from which this upgrade is supported.

- BMC Atrium Discovery 10.0
- BMC Atrium Discovery 9.x on RHEL 6

You can perform the following upgrade operations on a cluster or a standalone machine from the appliance’s UI:

- Upgrade BMC Atrium Discovery to a Patch, Service Pack or a version later than 10.2
- Upgrade the Operating System

To run an upgrade, you must be a user with admin security privileges such as the system user.

The preferred way to upgrade is through the UI. You can also use the tw_run_upgrade command line utility, though this is intended for use if there is an upgrade issue which cannot be resolved from the UI.

To prepare for the upgrade

1. Download the compressed upgrade archive from the BMC Electronic Product Distribution (EPD) site. See upgrade archive naming conventions
2. Upload the upgrade archive to the /usr/tideway/var/upgrade directory on any machine in the cluster or on the standalone machine. This can also be done through the upgrade UI if you are using a browser that supports HTML5 sufficiently.

⚠️ Note
To upload the upgrade archive using the UI, you must have sufficient space in 
/tmp, approximately 1.6GB.

Upgrade archive naming conventions

To help you locate the necessary file, the upgrade archives for BMC Atrium Discovery use the 
following file naming conventions. Product upgrades from BMC Atrium Discovery version 10 and 
later use a .tgz file extension. Previous versions used a .sh.gz extension and required manual 
verification using checksums. The compressed OS upgrade archive retains the .sh.gz file 
extension.

- For product upgrades later than 10.0:
  ADDM_Upgrade_Vn_nnnnnn_ga.tgz
  Where Vn is the BMC Atrium Discovery version number to which you are upgrading to and 
nnnnnn is the build number.

- For OS upgrades:
  ADDM_OS_Upgrade_64_6.yy.mm.dd_nnnnnn_ga.sh.gz
  Where yy.mm.dd is the date of the last package update from Red Hat and nnnnnn is the 
  build number.

Warnings

⚠️ Changes to OS Configuration Files

If you have made changes to OS configuration files on the appliance, these changes 
might be overwritten by the upgrade process. After the upgrade has completed, you must 
check any configuration files you have previously modified and reapply the changes as 
required.

⚠️ Database upgrade

This upgrade performs an upgrade of the BMC Atrium Discovery database. It is highly 
recommended that you do not skip running a backup. Where a backup is created, it can 
only be restored to an appliance running the pre-upgrade version.

⚠️ This upgrade includes significant changes to CMDB synchronization. In particular the 
filtering is changed. Many existing filters will not be compatible with the new filtering and 
will need to be recreated in the CMDB Sync filter tab after upgrade and before CMDB 
synchronization can be used.
For more information see, CMDB synchronization filters.

Upgrade considerations

Click the following items to show additional information or explanation.

For users with consolidating appliances

When a system uses consolidation, the suggested approach in this upgrade is:

1. Stop discovery on scanning appliances.
2. Ensure that all consolidation operations are complete.
3. Stop discovery on consolidating appliances.
4. Upgrade consolidating appliances.
5. Restart discovery on the consolidating appliances.
7. Upgrade scanning appliances as required.

Version 9.x scanning appliances can consolidate to version 10.x consolidating appliances, but version 10.x scanners cannot consolidate to earlier consolidating appliances. Once you have upgraded your consolidating appliances, you can then upgrade scanning appliances as required. You should definitely plan to upgrade them all eventually, it is not a permanent solution to leave some scanners at version 9.

When you run the upgrade, the timezone you have specified will be overwritten and returned to Europe/London unless you have updated the variable ZONE in /etc/sysconfig/clock. See Localizing the appliance for information on how to do this.

- The synchronization mechanism has changed in 10.2, to reduce the amount of querying performed on the CMDB. As a result it will be necessary to resynchronize a connection that existed prior to the upgrade. See Resyncing a CMDB Connection for more information.
- Where an upgrade makes changes to syncmapping files (see Default CDM Mapping and Syncmapping block), the initial CMDB syncs after upgrade might result in longer reconciliation times. Examples of such changes are key changes or attribute changes on a CMDB CI.

The following items are also affected by the upgrade:

- **SQL and JDBC credentials** — where SQL and JDBC credentials are in use, their existing properties files continue to be used and updated properties files are installed but not used. Where such credentials are unused, the old properties files are replaced with the latest.
You can re-run the upgrade to BMC Atrium Discovery 10.2 if it is terminated or fails. Destructive actions are recorded and are skipped on subsequent upgrade runs. Consequently when an upgrade is re-run, it is usually significantly quicker. You are unlikely to need to run the upgrade more than once, as it would only be required if an upgrade fails. See also the \texttt{tw_run_upgrade} command to re-run an upgrade.

The location used by the upgrade for temporary files must be readable by the \texttt{tideway} user. The default location (which will be created if it does not exist) is \texttt{/usr/tideway/tmp}, and can be changed using the command line option \texttt{--tmpdir}.

\textbf{To run the upgrade}

1. The upgrade places temporary files in \texttt{/usr/tideway/tmp}. If this directory already exists, make sure it is readable and writeable by the \texttt{tideway} user.
2. From the Appliance section of the \textbf{Administration} page, click \textbf{Upgrade}.

   \begin{center}
   \includegraphics[width=0.4\textwidth]{upgrade_screen}
   \end{center}

   \textit{This screen shows the Upgrade page.}

3. Ensure that there are no warnings in the Pre-Upgrade Check section. All pre-upgrade checks pass when the following required conditions for the upgrade are met:
   - (Applicable only in a cluster) All the members of the cluster are functioning.
   - Discovery is stopped.
   - The credential vault is open.
   - No discovery run or consolidation in progress.
   - No synchronization with BMC Atrium CMDB in progress.
4. Create a backup of the appliance or cluster before running the upgrade.
5. Copy the upgrade archive to the \texttt{/usr/tideway/var/upgrade} directory. This can be done through the upgrade UI if you are using a browser that supports HTML5 sufficiently.
   a. If the upgrade file is missing and your browser supports HTML5, you see an Upload button. Click this and choose the upgrade archive from your file system. Click upload to upload the archive. A progress bar shows the progress of the upload.
   b. If the upgrade file is missing and your browser does not support HTML5, the following message is displayed: "\texttt{No upgrade archive found. Please upload an ADDM upgrade archive to \texttt{/usr/tideway/var/upgrade}.} Once you have uploaded an archive, click \textbf{Check for upgrade archive}.\"
This screen shows Upgrade page where an archive has been uploaded.

6. At this stage you can choose the Advanced mode upgrade, to do this, continue this procedure from **Advanced mode upgrade**.

7. To perform the "no questions" upgrade click **Apply Upgrade**.
   If you are using a cluster, the archive is distributed to other members of the cluster. The upgrade runs and when it completes, the system reboots.

**Advanced mode upgrade**

In the final step of the previous procedure, you click **Apply Upgrade** to start a "no questions" upgrade. To perform an upgrade in which you control when the files are distributed and when the actual upgrade starts, select the Advanced Mode checkbox.

1. Click **Prepare Upgrade**.
   The upgrade files in the compressed upgrade archive are extracted and the UI displays the status in the progress bar and notifies you on completion.

2. (Only if you are upgrading a cluster) Click **Distribute Upgrade**.
   When the distribution starts, you can monitor the progress of the distribution for all the cluster members. Once the distribution is completed, you see a screen similar to this:

   ![Distribute Upgrade Screen](image)

   **This screen shows the upgrade procedure where the upgrade file has been distributed but has not yet run.**

3. Click **Run Upgrade**.
   A window shows the progress of the upgrade and details of the upgrade:
This screen displays the progress of the cluster upgrade from the UI. When the upgrade completes, the system reboots.

### Post upgrade messages

When the system reboots after an upgrade, some upgrade actions are carried out in the background. Some of these actions may require your intervention and those are listed as Post Upgrade Messages and can be viewed and dismissed by clicking the **User Notification Messages** icon in the dynamic toolbox. These messages are available only to users with admin security privileges and contain the following information:

- The components which have raised the messages.
- The corresponding severity level of each message.
  - Error — a high priority issue requiring immediate attention.
  - Warning — a medium priority issue which will require attention soon.
  - Information — a low priority issue of which you should be aware.
- A link to the UI page from where the issue can be resolved may also be displayed.

#### Example post upgrade messages

Depending on the configuration of the pre-upgrade system, some of the following steps may be displayed:

**CMDB synchronization filters**

In version 10.2 the CMDB synchronization mechanism has been changed and most filters created using previous versions will no longer work. The filters may be in one of the following states:

- Working
- Non functioning, but reviewable. These need to be recreated in the **CMDB Sync filter tab**.
- Removed. These too need to be recreated in the **CMDB Sync filter tab**.

For further information, see [Recreating legacy filters](#).

**Review patterns flagged by the upgrade**

The upgrade flags any pattern modules that refer to the old network model and therefore need changes to continue to work correctly. Flagged TKU patterns are addressed by activating the new TKU. However, flagged custom patterns require manual review to decide what changes are required.
The Knowledge management page lists pattern modules flagged by the upgrade. If no modules are listed then no action is required. Any listed modules show the errors and/or warnings generated by the upgrade when patterns were compiled. Depending on the severity of the compile messages, pattern modules can be left enabled or disabled. However, all messages should be reviewed to avoid future issues with your patterns. Pattern modules and packages should be updated in the usual way for your environment; either by editing via the user interface or by uploading a new version.

Any pattern packages that you do not wish to update immediately can be deactivated and will no longer be listed. Alternatively, any pattern packages that are no longer required can be deleted.

**Activate new TKU**

The upgrade installs a new TKU package (TKU-2015-04-2-ADDM-10.2+) but does not activate it. The new TKU must be activated before performing discovery. Information on activating and deactivating TPL packages is available here.

**Export mapping sets**

The upgrade checks whether there is a newer version of each of the installed mapping sets. If a mapping set has changed since the last version, either by the user modifying it or BMC releasing a newer version, then a warning is displayed to the user. The original mapping is renamed by the upgrade to append ".old" to the mapping set descriptor (the file ending with ".properties") and "_old" to the directory containing the mapping files. You can either:

- Ignore the warning if the export framework is not being used.
- Compare the old mapping set to the new one and keep the new one (i.e. do nothing).
- Compare the old mapping set to the new one and decide to keep the old one, in which case the user needs to manually delete the newer mapping descriptor and directory and rename the old ones (removing the .old and _old postfixes).
- Compare the old mapping set to the new one and merge the changes. If the changes to the mapping set have been performed by BMC Software then these changes will be listed in the release notes and the user can apply these changes manually to their own copy of the mapping set.

**Check Windows proxy compatibility**

<table>
<thead>
<tr>
<th>BMC Atrium Discovery 8.3 Windows Proxies</th>
</tr>
</thead>
<tbody>
<tr>
<td>The default keys and certificates used in very old versions of BMC Atrium Discovery expire in 2015. The transition to newer default keys started with version 9.0, meaning that, as long as the version 10.2 appliance is configured to use the legacy keys, an appliance can use a version 9.0 proxy, but it cannot use a version 8.3.x proxy that uses the default keys.</td>
</tr>
</tbody>
</table>
To use a version 8.3.x proxy, by far the best approach is to upgrade the proxy to version 10.2. If that is not possible, suitable CA certificates must be installed on the proxy.

To make 8.x proxies operate with a 10.2 appliance

To make 8.x proxies operate with a 10.2 appliance you must copy the new certificate authority file to the main proxy folder and each of the runtime folders. To do this:

1. Ensure that the appliance is using the **legacy key and certificate**.
2. Concatenate the two files containing the public keys, /usr/tideway/etc/ca/appliance_ca.pem (new public key) and /usr/tideway/etc/ca/appliance_ca_1.pem (old public key), using the following command:
   ```
   cat /usr/tideway/etc/ca/appliance_ca.pem /usr/tideway/etc/ca/appliance_ca_1.pem > /usr/tideway/etc/ca/appliance_ca.pem
   ```
3. Copy /usr/tideway/etc/ca/appliance_ca.pem from the appliance to the 8.x proxy folders. Put a separate copy in the main proxy folder and each of the runtime folders. For example:
   - C:\Program Files (x86)\BMC Software\ADDM Proxy\etc
   - C:\Program Files (x86)\BMC Software\ADDM Proxy\runtime\<proxy name1>\etc
   - C:\Program Files (x86)\BMC Software\ADDM Proxy\runtime\<proxy name2>\etc
4. Restart all of the proxies.
5. Move the appliance_ca.pem file out of the /usr/tideway/etc/ca/ directory, for example to the /tmp directory.

To make proxies upgraded from 8.x to 9.x operate with a 10.2 appliance

Proxies that have been upgraded from 8.x to 9.x also require a manual upgrade of their default certificates in order to operate with a 10.2 appliance. To do this:

1. Copy C:\Program Files (x86)\BMC Software\ADDM Proxy\etc\ca_01.pem into each of the proxy runtime folders. For example:
   - C:\Program Files (x86)\BMC Software\ADDM Proxy\runtime\<proxy name1>\etc
   - C:\Program Files (x86)\BMC Software\ADDM Proxy\runtime\<proxy name2>\etc
2. Restart all of the proxies.

Proxies deployed from a 9.x proxy manager, whether upgraded or newly installed, work without any manual configuration.

Check the Windows proxy **compatibility** matrix to determine whether you need to upgrade Windows proxies.
Baseline changes

The baseline tool tracks changes to the system configuration from a known baseline. After an upgrade, the appliance configuration will have changed significantly. You should view the baseline page after an appliance upgrade and examine the changes made to the system. When you understand the changes that have been made, you can rebaseline the appliance so that the tool can check for changes from the configuration after upgrading to the newer version of BMC Atrium Discovery.

Resolving upgrade issues

If you are using any version of BMC Atrium Discovery 10.1 or later, the preferred method for upgrading a Patch, Service Pack, or version is from the appliance's UI. For more information, see Upgrading BMC Atrium Discovery.

⚠️ Upgrading from an earlier version

There is no upgrade path from BMC Atrium Discovery version 7.5 to version 10.1. Rather, you need to migrate your data to a new installation of BMC Atrium Discovery version 9.0 and then upgrade to BMC Atrium Discovery version 10.1.

If you are upgrading from an earlier 7.X or 8.X version you will first need to upgrade to one of the versions listed above.

Resolving upgrade issues

You must be logged in to the appliance using a screen session. Enter:

```
[tideway@localhost ~]$ screen
```

See Recovering from a lost connection using screen.

In BMC Atrium Discovery 10.0 and later versions, you can perform the following upgrades on standalone machines and clusters either from the BMC Atrium Discovery UI or by running the `tw_run_upgrade` command line utility:

- Upgrade to a BMC Atrium Discovery Service Pack or a version later than 10.0
- Upgrade the Operating System

⚠️ The preferred way to upgrade is through the UI instead of the `tw_run_upgrade` utility. The utility is recommended only if there is an upgrade issue and you are required to resolve it by using the options provided by the utility.
**tw_run_upgrade capabilities**

The `tw_run_upgrade` utility is an interactive command line tool. Based on the upgrade issue you have, it informs you about the option you must run the utility with and any additional action you are required to perform to resolve it. To learn about some of the typical scenarios where you will use the `tw_run_upgrade` utility, see Resolving an incomplete upgrade process.

**Running `tw_run_upgrade`**

Before you start an upgrade, make sure that you have downloaded the compressed upgrade archive from the BMC Electronic Product Distribution (EPD) site and copied that to the `/usr/tideway/var/upgrade` directory of the machine from which you will run the upgrade.

To run the upgrade using the `tw_run_upgrade` utility, you must login to the machine from which you will run the upgrade as the root user and type the following command:

```
tw_run_upgrade [options]
```

where `options` are any of the options described in `tw_run_upgrade options` and the common command line options described in Using command line utilities.

**tw_run_upgrade options**

The following table describes the `tw_run_upgrade` utility options which are not listed in Using command line utilities:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--auto</td>
<td>Specifies to suppress prompts for confirmation of continuing the upgrade after time estimation.</td>
</tr>
<tr>
<td>--distribute</td>
<td>Specifies to prepare the upgrade and distribute the upgrade files over the cluster. It does not run the upgrade process. You can distribute the upgrade files first and run the upgrade later.</td>
</tr>
<tr>
<td>--fix-interrupted</td>
<td>Specifies to unlock the machine after a previous upgrade run was interrupted.</td>
</tr>
<tr>
<td>--no-version-check</td>
<td>Specifies not to check the upgrade compatibility.</td>
</tr>
<tr>
<td>--restart</td>
<td>Specifies to unlock the machine where the upgrade was interrupted and restart the upgrade on that machine.</td>
</tr>
<tr>
<td>--start</td>
<td>Specifies to prepare the upgrade, distribute the upgrade files (if not already distributed), and start the upgrade process. This is the default mode of execution.</td>
</tr>
<tr>
<td>--tmpdir=TMP_DIR_PATH</td>
<td>Specifies to set up the upgrade temporary directory. If a machine is likely to run out of disk space, you can set a symlink for the upgrade directory to a different mounted file system which has free disk space.</td>
</tr>
</tbody>
</table>
Resolving an incomplete upgrade process

The following section contains user examples of some typical scenarios where you will use the tw_run_upgrade utility in a standalone machine and a cluster:

User example for a standalone machine

If the upgrade process is interrupted on a standalone machine, the command line directs you to run the upgrade process again by running the `tw_run_upgrade --start` command.

User examples for a cluster

- **Using the `restart` command option:** While running the upgrade, if the upgrade process is interrupted on one of the machines in the cluster after the services were stopped, it puts that machine into a locked state and prevents the other members of the cluster (where the upgrade has completed) from starting. If you attempt to start the services manually from the command line of the machine where the upgrade was interrupted, it directs you to unlock it and resume the upgrade process by running the following command:

  ```
  sudo tw_run_upgrade --restart
  ```

  Running the command unlocks that machine and resumes the upgrade process. Once the machine is upgraded and it reboots, all the machines in the cluster will also start.

  **Warning**

  The upgrade must be restarted (`tw_run_upgrade --restart`) only if it fails at a stage during the actual execution of the upgrade script. If the upgrade fails at an earlier stage (for example, during preparing the upgrade or distributing the upgrade files on the cluster), restarting will only upgrade the machine from where you are running the upgrade.

- **Using the `--fix-interrupted` command option:** While running the upgrade, if the upgrade process is interrupted on one of the machines in the cluster before the services were stopped, it puts that machine into a locked state and displays an error message on the machine from where you are running the upgrade:

  ```
  Member no longer aware of ADDM Upgrade operation
  ```

  The interruption stops the services only on the machine where the upgrade was interrupted. If you attempt to start the services on that machine manually from the command line, it directs you to unlock it by running the following command:
sudo tw_run_upgrade --fix-interrupted

Running the command unlocks that machine. The interactive command line tool informs you if any additional intervention is required before you can run the --start option and run the upgrade again for that machine.

Post upgrade steps

⚠️ See post upgrade steps in Upgrading BMC Atrium Discovery.

As well as the notes on that page you should refer to the postupgrade_Vnn_TODO.log written out by the upgrade. This contains tailored advice of the tasks that must be completed on that particular appliance and these must be completed for correct future behavior.

Messages in the upgrade log

During the upgrade the firewall (iptables) is restarted. When a kernel upgrade is part of the upgrade, the firewall is unable to restart as there is a mismatch between the running kernel's version and the kernel on disk. The firewall logs a FATAL message, but as this is entirely expected, the upgrade wraps it in an information message:

```plaintext
```

This is expected behavior and does not indicate a problem with the upgrade.

Recovering from a lost connection using screen

If you lose the connection to the appliance during the upgrade and you have used screen, you can reconnect to the appliance and recover the virtual terminal running the upgrade. To do this:

1. Reconnect to the appliance and login as the tideway user.
2. List the current screen sessions. Enter:

   ```bash
   [tideway@appliance01 ~]$ screen -ls
   There is a screen on:
   23274.pts-0.appliance01 (Detached)
   1 Socket in /var/run/screen/S-tideway.
   ```

   You can re-attach to it with a simple command:

   ```bash
   [tideway@appliance01 ~]$ screen -r
   ```
The virtual terminal is recovered and you can see how the upgrade is progressing.

## Moving node keys

In BMC Atrium Discovery version 9.x, multiple appliances were required to discover large estates. Where multiple appliances synchronize to the CMDB, each appliance uses its own dataset, and the CMDB reconciliation rules merge the datasets into a golden dataset called BMC.ASSET.

With the introduction of clusters in BMC Atrium Discovery version 10, the entire estate may be discovered by a single cluster.

When you create a cluster, only one machine can be in the non-default state, that is, have any changed configuration or any data. You can upgrade one of the machines and retain its data, the others can be upgraded, but must be reset when you create the cluster. The data from the machines that are reset is removed and must be rescanned. The algorithm that identifies hosts (actually root nodes, which are Hosts, Network Devices, SNMP Managed Devices, Printers, Storage Devices, and Management Controllers) may come up with a different identifier or key each time a root node is newly created in the datastore. However, CMDB sync relies on the keys of the root nodes remaining the same; if they are not the same, then CMDB sync may create duplicate CIs.

A similar situation arises when the `tw_model_wipe` utility is used in a BMC Atrium Discovery system that synchronizes data to CMDB.

When a host is scanned successfully but subsequently cannot be reached, the host node is aged and deleted. On its deletion, an event is sent to CMDB and the corresponding CI is deleted. If `tw_model_wipe` is used, or the machine's configuration is and reset on joining a cluster, then the event is never sent. Consequently the CI representing the aging host is never removed.

In BMC Atrium Discovery version 10 a root node key information export utility is provided which enables you to extract information and the key of root nodes from the version 9.x appliances. You can then upload this to the version 10 appliance where it is stored in the datastore as RootNodeKeyInfo nodes.

When the version 10 appliance or cluster scans the estate, any potentially new root node is compared against the RootNodeKeyInfo nodes, and if a match is found for the discovery target, a new node is created using the key from the RootNodeKeyInfo node and the RootNodeKeyInfo node is deleted. If no match is found in the RootNodeKeyInfo nodes, then a new node is created with a new unique key.

### To move node keys in upgraded systems

1. Upgrade the first appliance to BMC Atrium Discovery version 10.
2. Download the utility from the appliance you just upgraded. The utility is contained in a zip archive which is available from the Tools tab on the Discovery page.

3. Upload the archive onto the other BMC Atrium Discovery version 9.x appliances that will be reset in the planned cluster.

4. Extract the utility. Enter:

   ```
   [tideway@appliance01 ~]$ unzip root_node_key_export.zip
   ```

5. Run the utility on each appliance to create the XML key information file. Use the appliance name as part of the filename. For example, to export appliance01 keys:

   ```
   [tideway@appliance01 ~]$ tw_root_node_key_export appliance01_key_info.xml
   ```

6. Copy the XML files onto BMC Atrium Discovery version 10 appliance and run the import utility. For example, to import appliance01 and appliance02 keys into appliance100:

   ```
   [tideway@appliance100 ~] $ tw_root_node_key_import appliance01_key_info.xml appliance02_key_info.xml
   ```

7. Upgrade the remaining version 9.x appliances to BMC Atrium Discovery version 10 and reset their configuration.

8. From the appliance that you imported the files, create a cluster adding the other upgraded appliances as members.

**Import warnings**

When you run the node key importer, you may encounter warnings. For example:

- If the appliances had overlapping scan ranges.
- A host key was contained in the import data and you had already scanned the host and generated a new key.

These warnings are also logged in `$TIDEWAY/log/tw_root_node_key_import.log`. For example:

```
test.xml: Imported 2 root node keys.

Information for 4 root nodes was not imported because they match the following existing nodes:

  sw-london-01.bmc.com (kind: NetworkDevice, key: GIvkT/7rwRw62h3crLoKf==)
  London (kind: SNMPManagedDevice, key: V1U+clotkQr3GAaMU9u5Q==)
```
BMC-Main-Printer (kind: Printer, key: BT2Veaa2IQFuelbAVUs6Xw==)
localhost (kind: Host, key: dEFkpJtnDbErqt5iOWImlg==)

Imported information for 1 root node matches an existing root node information:

Existing node: mldev (root_node_kind: Host, key: 12345)
Imported node: mldev (root_node_kind: Host, key: 1234)

In this run, six RootNodeKeyInfo nodes were considered. The first two RootNodeKeyInfo nodes were imported, no additional information is logged on this. The next four were not imported because they match nodes that are already in the datastore.

There is also a warning that one of the imported RootNodeKeyInfo nodes matches another RootNodeKeyInfo node that has previously been imported. However, it is not identical; the key is different. The RootNodeKeyInfo node is imported as it is unclear whether they refer to the same host. Two appliances may have had overlapping scan ranges and both scanned the same host.

Unused imported keys
You can view a report on unused imported keys. In the Atrium Discovery Deployment section of the main Reports page, the Root node key information which has not been used provides this information.

To preserve IDs when using tw_model_wipe
The tw_model_wipe utility deletes the contents of the datastore. See the documentation before attempting to use it. On the BMC Atrium Discovery version 10 machine from which you plan to delete contents of the datastore:

1. Run the utility to create the XML key information file. Use the appliance name as part of the filename. For example:

   [tideway@appliance01 ~]$ tw_root_node_key_export appliance01_key_info.xml

2. Run the tw_model_wipe utility. For example:

   [tideway@appliance01 ~]$ tw_model_wipe --force

3. Run the import utility.

   [tideway@appliance01 ~]$ tw_root_node_key_import appliance01_key_info.xml
When you rescan your estate, any potentially new root nodes are compared against the RootNodeKeyInfo nodes, as described earlier. Previously scanned root nodes are recreated using the key from the RootNodeKeyInfo node and the RootNodeKeyInfo node is deleted. On subsequent CMDB synchronizations the correct CIs are updated.

Configuring after upgrade

The post-upgrade activities that you might consider are the same as post-installation activities. If you have already performed these configuration steps, they are retained on upgrade and do not need to be repeated.
Using

The following topics introduce the standard pages and methods of navigating the BMC Atrium Discovery UI:

- Navigating the BMC Atrium Discovery UI
- Performing basic tasks
- Configuring discovery
- Running discovery
- Discovery targets
- Improving discovery
- Monitoring appliance performance
- Knowledge management
- Importing data
- Managing your IT infrastructure
- Managing your business applications
- Auditing a data center
- Mapping your business applications
- Managing reporting
- Using the Search and Reporting service
- Exporting data

Navigating the BMC Atrium Discovery UI

BMC Atrium Discovery uses a web-based thin client interface. No software is installed on a user's workstations. The following topics explain how to navigate the BMC Atrium Discovery UI:

- Supported browsers
- Page layout of the UI
- Tips for navigating the UI
- Downloading user documentation

Supported browsers

On such browsers, we have thoroughly tested the functionality and will consider it a bug if something does not work. These are the browsers we recommend using.

- Firefox 24.0 and above
- Chrome latest version
- Internet Explorer 8 (The Software Context view is not supported on Internet Explorer 8)
- Internet Explorer 9 (The Software Context view is not supported on Internet Explorer 9)
- Internet Explorer 10 (Desktop mode only, not Windows UI/Metro mode - see below)
- Internet Explorer 11 (Desktop mode only, not Windows UI/Metro mode - see below)
**Internet Explorer Compatibility Mode**

Internet Explorer Compatibility Mode is not supported, as this forces the browser to behave as an older (unsupported) version. This behavior can be controlled with the Internet Explorer settings:

- Tools > Compatibility View
- Tools > Compatibility View Settings

You should use a minimum screen resolution of 1024x768.

**Page layout of the UI**

Your view of the UI and the options that you can access are dependent on your access privileges. Your access privileges are defined by the user groups that you are a member of. See your BMC Atrium Discovery Administrator for details of your privileges.

A typical BMC Atrium Discovery page contains the following major sections:

- Version number and search bar
- Primary navigation bar
- Dynamic secondary navigation bar
- Dynamic toolbox
- Actions, reports, and visualizations

These major sections are described in the following sections.

**Version number and search bar**

The BMC Atrium Discovery version number and the search box are displayed at the top-right of each page:

This screen illustrates the version number and location of the search bar.

Icons that display a menu when clicked have a down arrow in the bottom corner. Click the Search Options icon to display the Search dialog where you can tailor your searches, as illustrated in the following screen.
This screen illustrates the Search dialog and options for refining a search.

The search options menu enables you to search the database for a keyword. To close this menu, click the Search Options icon again or alternatively, click the Close button in the top-right corner of the menu. For more information about the available search options in BMC Atrium Discovery, see Searching for data.

Primary navigation bar

The primary navigation bar enables you to access the main functional areas of BMC Atrium Discovery.

This screen illustrates the main functions on the primary navigation bar.

The primary navigation bar contains the following major tabs:

- **Home** — Returns you to your home page. See Viewing the home page for more information.
- **Applications** — Enables you to view, search and set up your organization's supported applications that have been modeled from a business perspective. This includes imported or created data about applications supplemented by additional logical information such as business location and ownership information. See Managing your business applications.
- **Infrastructure** — Enables you to view, search and set up your IT infrastructure: the basic hardware (host computers) and Software Instances (running programs) in your organization. This information is normally discovered by BMC Atrium Discovery from your environment but it can also be imported information. See Managing your IT infrastructure.
- **Discovery** — Enables you to view, search, set up and amend your Discovery status. You can also manage your credentials, patterns and Discovery reports from this tab. For a full description of the Discovery tab, see Chapter 6 of the BMC Atrium Discovery Configuration Guide.
- **Reports** — Enables you to run a number of reports on the data in your system. See Managing reporting.
- **Administration** — Enables you to view, search, set up and amend your configuration and management options; many of which are restricted to users with Admin access rights. See Navigating the administration interface.

**Dynamic secondary navigation bar**

The dynamic secondary navigation bar is populated with a number of different buttons depending on the tab displayed in the primary navigation bar. The default view for the primary navigation bar tab can also be accessed by clicking the button to the far left on the secondary navigation bar.

For example, when you click the **Discovery** tab on the primary navigation bar, the following buttons are displayed:

![Dynamic secondary navigation bar](image)

This screen illustrates an example of the tabs on the dynamic secondary navigation bar.

For this example, the dynamic secondary navigation bar contains the following major tabs:

- **Discovery Status** — Displays the Discovery Status page. This is the default view.
- **Credentials** — Displays the Login Credentials page where you can configure the login credentials for discovery targets.
- **Pattern Management** — Displays the Pattern Management page where you can upload, download, activate and deactivate patterns.
- **Integration Points** — Displays the Integration Points page where you can manage connections and queries to databases.
- **Discovery Reports** — Displays the Discovery Reports page where you can view and control the discovery process, view Directly Discovered Data, and report on Discovery Accesses.

**Dynamic toolbox**

A dynamically populated toolbox is located below the primary navigation bar.

![Dynamic toolbox](image)

This screen illustrates the icons on the dynamic toolbox.

**Default icons**

By default, the dynamic toolbox contains the following icons from left to right:

- **Available Dashboards** — Displays a drop-down list showing the dashboards that are available. See Dashboards for more information.
Groups — Displays the Groups drop-down so that you can manage host groups. See Manual grouping for more information.

User
- General Preferences — Displays the general user preferences page.
- Application Preferences — Displays the application preferences page.
- Change Password — Displays the change password page.

Icons that show a menu when clicked have a down arrow in the bottom-right corner. Click the User icon to display the dialog.

Appliance Status — Displays the appliance status page. Click the Appliance Status icon to show the dialog box.

This screen illustrates the current status of the appliance. To configure the Appliance Status, see Appliance status.

Recent Items — Displays the Recent Items list. Click the Recent Items icon to show the dialog box.

The recent items menu lists your ten most-recently accessed pages, enabling you to navigate back to them easily. This list is retained between sessions. Click Clear to clear the list. When items are deleted from the system, they are not removed from the recent items list.

Help
- Documentation — Displays the contents page of the BMC Atrium Discovery user documentation available in this release. See Downloading user documentation.
- Support & Community — Displays the Community page of the Tideway website. From here you can contact Customer Support or access the current BMC Atrium Discovery online documentation and other information about the company.
- About — Displays information about this version of BMC Atrium Discovery. Click the Help icon to display the dialog box.

Dynamic icons
The Dynamic toolbox also contains icons which are displayed on other nodes, such as when you are viewing the details of a host. The following icons are provided:

Attachments — Lists the attachments attached to this node and enables you to manage them. For details, see Managing attachments. Click the Attachments icon to display the dialog box.
• **Discovery Session Logs** — Displays a list of Discovery session logs. This is only available for host nodes.

  Click the Discovery Session Logs icon to display the dialog box.
  Select a session log from the list and click the link. For more information about Discovery Session Logs, see Appendix A in the BMC Atrium Discovery Configuration Guide.

⚠️ **Note**

  The Session Logs icon is not available, or has been disabled, if there are no logs associated with the host you have selected.

• **Additional Links** — Lists any external links associated with this node. For example, the drop-down dialog below provides links to different search engines which might provide further useful information about this node.

  Click the Attachments icon to show the drop-down dialog.

For information about configuring external links, see [Configuring additional links](#).

### Actions reports and visualizations

A cluster of drop down menus is provided on node pages or node lists. These are Actions, Reports, and Visualizations. On node pages only, a Related drop down is also available.

#### Actions

The Actions drop down menu is dynamic and changes depending on the page on which it is displayed. On a Host List page, the Actions drop down menu has the following options:

- Only show selected: Filters the list to only show those nodes that are selected.
- Export as CSV: Exports the selected nodes as CSV files.
- Scan: Performs a snapshot scan of the selected nodes.
- Create Host Profiles: Creates Host Profiles of the selected nodes.
- CMDB Sync: Syncs the selected nodes to all connections or one of the listed connections.
- CMDB Sync Preview: Shows a software context preview of the selected node and connected nodes as synced to the CMDB.
- Manual Groups: Opens the manual grouping menu.
- Destroy: Destroys the selected nodes.

#### Reports

The Reports drop down menu is dynamic and changes depending on the page on which it is displayed. The Reports drop down contains a list of reports which might be run for ALL items in the list (which might be on multiple pages), or the selected items.
To run a report for all items in a list page, ensure that no items are selected by clicking the None selection option on the left side of the page. Then from the Reports menu, select the report that you want to run.

To run a report for the selected items in a list page, select the report from the Reports menu.

Visualizations

The Visualizations menu is dynamic and changes depending on the page on which it is displayed. The Visualizations menu contains a list of reports which might be run for ALL items in the list (which might be on multiple pages), or the selected items.

To run a visualization for all items in a list page, ensure that no items are selected by clicking the None selection option on the left hand side of the page. Then from the Visualizations drop down, select the visualization that you want to run.

To run a visualization for the selected items in a list page, select the report from the Visualization menu.

Related

The Related drop down menu is dynamic and changes depending on node type of the page on which it is displayed. The Related menu enables you to navigate around the model by traversing from a node to any other related node. The drop down displays the number of related nodes for each target node, and where different relationships exist to the same node kind, the label is shown to differentiate between them, for example IPv4 and IPv6 addresses. Click the required item to navigate to its page, or to a list page where there are multiple items linked by the same relationship.

On a Host page, the Related drop down menu has the following options:

- Packages
- File Systems
- Hosts
- Network Interfaces
- Software Instances
- via IP Address
- IP Addresses (IPv4 Addresses)
- IP Addresses (IPv6 Addresses)
- Runtime Environments
- Endpoint
- Network Device via Network Interface > Network Interface
- Software Instance (Containing VM)
- Directly Discovered Data
- Device Info
- Discovered Packages
- Discovery Access
- Host Info
Tips for navigating the UI

- Choose one of the major tabs on the primary navigation bar to access one of the main sections of BMC Atrium Discovery (Applications section, Infrastructure section, Discovery section, Reports section, Change section, or Administration section).
- The top-level page for each section displays a summary of the objects in that section.
- To search the system, use the Search option in the Version Number and Search Bar. For more information, see Version Number and Search Bar and Searching for data.
- Most reports available appear in the Reports section. You can also run some of the most common reports from the BMC Atrium Discovery Home Page and from the top-level page of the Applications and Infrastructure sections. There are also context-sensitive reports available for some individual nodes.
- When you run a search or a report, items that meet your criteria are listed in the main display window.
- Items that appear as a highlighted hyperlink on any screen indicate objects in the database that you can view in detail. Click any of these to display the relevant detail page. See Using View Object pages.
- A progress bar can be set up to appear while navigating between different screens. The progress bar will only appear after the number of seconds that you specify. See Setting Your Application Preferences.

Downloading user documentation

From the BMC Atrium Discovery product UI, you can access the online product documentation and download the available documentation as PDF. The online documentation may be more up to date than the PDF version.

1. Click the Help icon from the dynamic toolbox.
2. Click Documentation.
   The full list of available BMC Atrium Discovery online user documentation and a link to download the PDF version are displayed.
3. If you are connected to the Internet and want to view the online documentation, select a document from the list and click on the online documentation link (for example, Getting started).
4. If you want to download the available PDF documentation, click on the PDF link. A popup screen enables you to open or save the document.

Performing basic tasks

The following topics introduce the basic tasks in the BMC Atrium Discovery:

- Setting preferences
- Viewing the home page
• Viewing summary list pages
• Using View Object pages
• Searching for data
• Comparing the history of nodes
• Viewing dependency visualizations
• Managing attachments

Setting preferences

To set application preferences

1. Click the User icon from the dynamic toolbox on the right side of the screen, below the primary navigation bar.
2. Click the Application Preferences link from the drop-down list.
3. From the Application Preferences page, enter your preferred settings in the appropriate fields.
   • Select the date format to be displayed in either UK or US style.
   • Specify the number of search results to display per page. For more information about running searches, see Searching for data.
   • Specify the size of the Recent Items list. For more information about recent items, see Default Icons.
   • Select the History view type. This can be Comparative or Raw. For further information about the display of history, see Comparing the history of nodes.
   • Select the default delimiter for exported data. This can be a Comma, a Semicolon, or a Tab.
   • Select whether to include a byte-order-mark (BOM) in CSV export files. Including the BOM is required for correct handling of multibyte characters in some applications.
   • Select the XML version required for reports. This can be 1.0 or 1.1.
   • Select the default paper size for PDF reports.
4. After you have entered your settings, click Apply.

To set general preferences

1. Click the User icon from the dynamic toolbox on the right of the screen, below the primary navigation bar.
2. Click the User Details link from the drop-down list.
3. From the Actions drop down on the Person page, select Edit.
   The page is refreshed with the following editable fields:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this user.</td>
</tr>
<tr>
<td>Phone</td>
<td>Telephone contact number for this user.</td>
</tr>
<tr>
<td>E-Mail</td>
<td>Email address for this user.</td>
</tr>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>User name</td>
<td>User name of this user.</td>
</tr>
<tr>
<td>Employee ID</td>
<td>The employee ID number for this user.</td>
</tr>
<tr>
<td>Role</td>
<td>Role of this user.</td>
</tr>
<tr>
<td>Manager</td>
<td>Manager of this user.</td>
</tr>
<tr>
<td>Manager of</td>
<td>Personnel that this user is the manager of.</td>
</tr>
<tr>
<td>Home Location</td>
<td>Standard home location of this user.</td>
</tr>
<tr>
<td>Current location</td>
<td>Current location of this user.</td>
</tr>
<tr>
<td>Software Families</td>
<td>Relationship that defines the product families that you are a member of.</td>
</tr>
<tr>
<td>Business Owner (Hosts)</td>
<td>Relationship that defines the person responsible for this application</td>
</tr>
<tr>
<td>Business Owner (Application Instances)</td>
<td>Relationship that defines the person responsible for this application from a business perspective. Search for and select one Person object.</td>
</tr>
<tr>
<td>IT Owner (Hosts)</td>
<td>Relationship that defines the person responsible for this application</td>
</tr>
<tr>
<td>IT Owner (Application Instances)</td>
<td>Relationship that defines the person responsible for this application from an IT perspective. Search for and select one Person object.</td>
</tr>
<tr>
<td>Support Manager (Hosts)</td>
<td>Relationship that defines the support manager responsible for this</td>
</tr>
<tr>
<td>Support Manager (Application Instances)</td>
<td>Relationship that defines the support manager responsible for this application. Search for and select one Person object.</td>
</tr>
<tr>
<td>Completeness Issues</td>
<td>Any data quality issues. Optional.</td>
</tr>
</tbody>
</table>

4. After you have entered your settings, click **Apply**.

**Viewing the home page**

The BMC Atrium Discovery home page displays predefined dashboards that provide a consolidated overview of certain aspects of your modeled IT infrastructure. Dashboards are particularly useful for highlighting standards and exceptions. For example, you can highlight under-used servers, or a breakdown of the operating systems on which Apache webservers are running.

When you initially log on to BMC Atrium Discovery, the default dashboard screen displays.
The default dashboard contains a Welcome to ADDM channel which describes the new features in the release. Should you eventually tire of reading about the new features, you can dismiss the channel by clicking Dismiss. This is a per-user setting, so dismissing the channel will not deprive other users of this happy reading experience. Unfortunately, you cannot restore your own Welcome to ADDM channel once it is dismissed.

On subsequent logins, the most recently used dashboard is displayed. On any user-defined dashboards, each section can be selected by clicking and holding the heading bar and can then be moved around the page. When you move the section to a new position, the outline is moved and the other sections rearranged to accommodate the new position.

You can also remove a section by clicking the close icon in the top right corner of the section. You can refresh the section with the refresh icon to the left of the close icon. The changes that you make are immediately available to all other users.

You can configure the dashboards to provide your own custom home page. The dashboards provided are described in Types of dashboards, and their use and how to customize them is described in Using and customizing dashboards.

The standard installed dashboards cannot be edited, though they can be copied, and the copy can be edited.

**Types of dashboards**

The following dashboards are supplied by default and are described in the following sections:

- Default
- Baseline
- CMDB Sync Status
- Discovery
- Hardware Reference Data
- Host Network Summary
- Host Overview
- OS Product Lifecycle Analysis
Operating Systems
Server Consolidation
Software products - Apache Webserver
Software products - Oracle
Software Products Lifecycle Analysis
Software Products Summary
Virtualization

Default

The Default dashboard provides a concise summary of BMC Atrium Discovery's view of your IT infrastructure.

- Discovery Status
- Automatic Grouping
- Community Update
- Quick View

Baseline

The Baseline Dashboard provides the following information and reports to help you in baselining a data center:

- Infrastructure summary
- Software Products By Publisher
- Host count by OS Class
- Average number of Network Connections
- Automatic Grouping
- Baseline Support
- Host OS distribution
- Virtual vs. Physical

CMDB Sync Status

The CMDB Sync Status dashboard provides information about the CMDB Synchronization process. For each connection (CMDB target) the following information is displayed:

- Connection name
- Status
  - OK
  - Error
  - Paused (blackout window)
  - Paused (manually)
  - Disabled
- Continuous sync
  - On
  - Off
• Devices in queue

Click on any row to see the CMDB Connection Status page for that connection.

**Discovery**

The Discovery dashboard provides the following information about the current status of the results of discovery:

• Discovery Radar
• Current UNIX Access
• Current Windows Access
• Discovery Dashboard Reports
  
  Click the "Discovery is currently" link by the green RUNNING or red STOPPED button to display the Discovery page.

**Hardware Reference Data**

The Hardware Reference Data dashboard provides the following information about the physical and thermal characteristics of the hosts in your IT Infrastructure:

• Host Compute Power
• Host BTU/h per U size
• Host Compute Power and Watts
• Hardware Analysis Reports
• Host Watts per U size

**Host Network Summary**

The Host Network Summary dashboard provides the following information about the network connectivity of hosts in your IT Infrastructure:

• Average number of Network Connections
• Average number of Listening Ports
• Hosts By Network Interface Count
• Host Resiliency - Services
• Network Policies

**Host Overview**

The Host Overview dashboard provides the following information about the numbers of hosts, distribution of host types, and the memory and compute power of hosts in your IT infrastructure. Compute power is a rough guide to the power of a host and is determined by multiplying the clock speed by the number of logical CPUs.

• Host count
• Hosts by Type
• Host Compute Power and RAM
OS Product Lifecycle Analysis

The OS Product Lifecycle Analysis Dashboard provides the following information about the OS breakdown on the hosts in your IT Infrastructure. It also provides information about the proximity to those operating systems end of life.

- Product Lifecycle Risk - OS
- UNIX Operating Systems
- Common Reports
- Product Lifecycle Analysis
- Hosts By OS End of Life Date
- Windows Operating Systems

Operating Systems

The Operating Systems dashboard provides a breakdown of OS types and versions as well as links to detailed reports on patches and kernels.

- Microsoft Windows Operating Systems
- UNIX Operating Systems
- Operating Systems
- Host Resiliency - Kernel

Server Consolidation

The Server Consolidation dashboard highlights the hosts in your environment which are candidates for consolidation. This is particularly useful when planning a data center consolidation project as it highlights the areas in which you can make the greatest gains.

It provides charts showing breakdowns of the following:

- Host CPU speed and count
- Host Compute Power
- Hosts By Vendor
- Host Type by RAM in GB

Software products - Apache Webserver

The Apache Webserver dashboard provides information about all of the Apache Webservers in your IT infrastructure.

It provides charts showing breakdowns of the following:

- Apache Webserver Versions
- Servers running Apache: By Operating System
- Servers running Apache: Virtual vs. Physical
- Servers running Apache: Compute Power and RAM
Software products - Oracle

The Oracle dashboard provides the following information about all of the Oracle databases in your IT infrastructure:

- Oracle Versions
- Servers running Oracle: By Operating System
- Servers running Oracle: Virtual vs. Physical
- Servers running Oracle: Compute Power and RAM

Software Products Lifecycle Analysis

The Software Product Lifecycle Analysis Dashboard provides information about the software product in your IT Infrastructure. It also provides information about the proximity to those software products end of life.

It provides the following sections:

- Product Lifecycle Risk - Software
- Software Products By Category
- Common Reports
- Product Lifecycle Analysis
- Software By End of Life Date
- Software Products By Publisher

Software Products Summary

The Software Products Summary dashboard provides a breakdown of software products identified on your network, the categories of software products identified, and database versions. It also has links to detailed reports on Data Centre Standardization Software, Software Inventory, and Database Reports.

- Software Products By Publisher
- Software Products By Category
- Database versions
- Data Centre Standardization Software
- Software Reports
- Database Reports

Virtualization

The Virtualization dashboard provides an overview of software running on virtual machines, and the physical hosts on which the virtual machines are hosted. An example Virtualization dashboard is shown in the following screen:
This screen illustrates a Virtualization dashboard.
This dashboard provides charts showing breakdowns of the following:

- Windows OS Virtualization
- UNIX OS Virtualization
- Virtualization Reports
- Virtualization Technologies

Using and customizing dashboards

The following sections provide information about using and customizing dashboards:

- To view other dashboards
- To create a dashboard
- To edit a dashboard
- To add, rename, or remove a channel to an editable dashboard
- To move a channel

To view other dashboards

You can view any of the dashboards configured on your appliance by clicking the Available Dashboards icon from the dynamic toolbox on the Home page and choosing a dashboard from the menu.

The content in the section shown on the home page is referred to as a channel. There are a variety of types of content available for use in channels. For example, you can configure video content, web feeds, RSS feeds, in addition to the standard BMC Atrium Discovery reports. For example:

- Web feed channel
This screen illustrates an example of a Web feed channel.

- **Video channel**

This screen illustrates an example of a video channel.

Creating channels

Channels are created using the Administration > Channels page.

To create a dashboard

To add a new dashboard, select the Copy current Dashboard link from the Dashboard drop-down panel. A copy of the dashboard is saved, and it displays an Edit link enabling you to add or remove sections and rename it as required.

⚠️ **Channels on Reports Pages**

You can include any of the other channels on the report page in a dashboard. For example, you could add a few favorite reports to your dashboard. See Reporting basics for a list of reports that can be added.
To edit a dashboard

To edit a dashboard, select the **Copy current Dashboard** link from the Dashboard drop-down panel or select an existing copy of the dashboard from the Dashboard drop-down panel. The copy of the dashboard displays an **Edit** link which enables you to add or remove sections and rename it as required.

Changes you make to the layout or content of a dashboard are immediately available to all users. The dashboard that you are viewing is saved in your session and is displayed the next time you log on.

To add, rename, or remove a channel to an editable dashboard

1. Click the **Edit** link in the main page title.

   ![Selector panel illustration](image-url)

   **This screen illustrates a selector panel where you can edit channels.**

2. On the channels selector panel, rename the dashboard by using the **Dashboard Title** field. This renames the existing dashboard rather than saving a copy under a new name.
3. Click **Commit** to change the name.
4. Select or deselect a channel to add it to or remove it from the dashboard.

Changes that you make are immediate and are available to all other users.

To move a channel

You can move a channel in the default view of an editable dashboard by dragging and dropping it.

1. Move the cursor over the header of the channel. The cursor changes to a four pointed selector.
2. Drag the channel to the new location and drop it in the correct place.

The changes that you make are immediate and are available to all other users.
Viewing summary list pages

When you run a search or a report, summary information is displayed in the form of a list.

- Objects of different kinds
- Objects of one kind
- To filter a column
- To chart a column
- What you can do from a summary list page

Objects of different kinds

If objects of several different kinds are returned as a result of a search, the page that is displayed lists the number of objects of each kind (for example, searching for any attribute that contains the word ‘discovery’, as illustrated in the following screen.

Click any listed object kind to display summary details of all the objects of that kind.

Objects of one kind

If objects of one kind are returned as a result of a search, they are listed in summary form.

Only the Summary attributes of each object are listed. The number of matching items is displayed at the top of the list.

To filter a column

When you are viewing a result set you can filter individual columns to display different subsets of the full list. This enables you to easily refine the results from any result set for any displayed column.

1. Select a result set in BMC Atrium Discovery.
2. Click the **Filter Column** icon in any column.
   
   A screen shows up to ten of the most frequent values for the selected column. If more than ten values are present in the column, as in the following example, a filtered subset named Others is displayed in the screen.

![Filter Column Example](image)

3. Select which results you want to show and click that filtered link.
   
   The results are displayed for that filtered selection, as illustrated in the following screen.

![Filter Results Example](image)

This filtering method can be applied to any column in the BMC Atrium Discovery UI which has the column filter icon contained in the column header. You can also filter results sets which have already been filtered.

**To chart a column**

When you are viewing a result set in BMC Atrium Discovery, you can chart data directly from any column in that result set. This enables you to view a graphical representation of the data contained in any column showing the frequency of the data items for that column.

1. Select a result set in BMC Atrium Discovery.
2. Click the **Column Chart** icon in any column.
   
   A screen displays, showing the list of available charts (currently set for pie, bar and column charts).

![Column Chart Options](image)

3. Select the type of chart you want to show and click that chart link.
   
   The results are displayed for that chart (for example, the Pie Chart result for the Hardware Vendor column, as shown in the following illustration).

![Pie Chart Example](image)
If the maximum number of default data points is exceeded for any type of chart, smaller items are merged together to form an Others section in that chart. The limit for all three chart types is 20 data points.

For a full description of the BMC Atrium Discovery reporting capabilities, see Managing reporting.

What you can do from a summary list page

- If the displayed list extends over multiple pages, you can navigate through the list using the arrow buttons to display next and previous pages. You can enter the number of a page and click Go to display that page. If you enter a signed number, such as +3 or -5, you can move forwards or back the specified number of pages.
- You can sort the items in the list by clicking on the column heading you want to sort by. Items are sorted in ascending order; click again to re-sort in descending order.
- You can select one or more items and add them to the basket.
- Certain items allow you to perform specific actions, for example:
  - Rescan a host.
  - Select Hosts to generate a Host Information Document.
  - Rescan a Discovery Access.
  - Rerun Discovery Run.
  - View a Software Context for selected items.
- Click any entry in the list to display the relevant View Object page. See Using View Object pages.
- If the list page is the result of a search, you can refine the search to reduce the number of items returned. See Searching for data for more information.

Customizing list views

The default list view shows fixed columns for each list type and provides basic filtering on each column. A Query Builder on list views enables you to create complex queries through the UI (rather than the Using the Search and Reporting service) and immediately see the results in the list.

- Customizing columns
- How the three pane selection tool works
- How the column list works
- To filter multiple results in a column

Customizing columns

The default list view for each node type shows pre-defined columns. You can customize these views using the Customize Columns tab of the Query Builder. For example, the default list view for hosts shows the Virtual column which might not be useful to you.
How the three pane selection tool works

The three pane selection tool enables you to select an attribute to add as a column, or follow relationships to other nodes from which you can select attributes to add as a column. Relationships are displayed with a following right arrow (>). Scroll down the left hand pane and select the attribute that you require.

Search in the left hand pane of the three pane selection tool for the attribute that you wish to add. You can use the lookup tool, or scroll through the list. Additionally, beneath each pane is a lookup tool into which you can enter text to locate attributes by name. When you enter text a drop down list of matches is displayed, from which you can select the attribute or relationship to use.

- Clicking an attribute, for example Discovered OS Class, adds a Host: Discover OS Class entry to the column list.
- Clicking a relationship, for example Software Instance: Software Instances running on this host, displayed with a following right arrow (>), populates the next pane in the selection tool with the attributes and relationships of the destination node.
- Clicking a relationship in the second pane populates the third pane in the selection tool with the attributes and relationships of the destination node in the same way as before.
- Clicking a further relationship populates a fourth pane and scrolls the previous panes to the left, hiding the first pane. You can scroll back by clicking the arrow to the left of the selector panes.

How the column list works

The column list is simply a list of columns displayed in the order that they appear in the list view. In this view they can be reordered by dragging and dropping. An edit icon and a delete icon are also provided for each column.

To see the results of your changes, click Refresh results.
To cancel all changes, click Cancel.

To remove a column

To remove a column, click the cross icon relating to that column. The column label is deleted immediately. Click Refresh results to remove the column.

To edit the label on a column

To edit the label on a column, click the pencil icon relating to that column. The column name is redisplayed in a text edit field. Edit the name and click Refresh results to see the change.

To add a column

To add a column, click the one you want to add by using the three pane selector.
You can drag the column to the required position in the list. The attribute name (the column label) is displayed as editable text so you can change it if required. Click Refresh results to add the column.

To reorder the columns
To reorder the columns, drag and drop them as required.

To filter multiple results in a column
Filtering operates per row rather than on the individual results in that row. This means when you filter on a column which contains multiple results, all results are still displayed per row. However, each row contains the item that you filtered on.

For example, when you add a column containing the name of a software instance running on a host, each row contains a list of software instances running on that host. Filtering on a software instance results in all rows containing that software instance being displayed, even though that software instance is part of a list. The following screenshots show a Software Instance column which contains multiple software instances per host.

When this is filtered, the number of hosts reduces, though there are still multiple software instances per host.

This screen illustrates the results of filtering multiple software instances per host displayed in a Software Instance column.

Query Builder
The Query Builder enables you to select attributes on a node, and to group and evaluate conditions on those attributes or groups of attributes. You can follow relationships from the node and group and evaluate attributes on destination nodes in the same way. This enables you to create a list of nodes fulfilling those criteria. For example, you may create a list of hosts running web servers. Additionally, you can traverse from a node, using relationships, to other node types, enabling you to create lists of other nodes which are related to the original nodes. For example, starting from the list of web servers mentioned above, you may create a list of clusters containing web servers. The example below shows this.
The attributes that you select are not limited to those in the taxonomy; those created from TKU or your own patterns can be used. To make it easier to find the correct attributes, relationships, and traversals, only the most commonly used are displayed initially. You can reveal all for a node by clicking the **Show All** link.

To view the Query Builder, click the Customize control under the page title on a list view page.

- Three pane selection tool
- How traversal selection works
- To build a query

**Three pane selection tool**

The three pane selection tool enables you to select an attribute, or follow relationships to other nodes from which you can select attributes. Relationships are displayed with a following right arrow (→). Scroll down the left hand pane and select the attribute that you require.

Search in the left hand pane of the three pane selection tool for the attribute that you wish to select. Initially the tool displays most commonly used attributes to make it easier to find the one you need. Click **Show All** to see all attributes on the node. You can scroll through the list or use the lookup tool beneath each pane. When you enter text into the lookup tool a drop down list of matches is displayed, from which you can select the attribute or relationship to use.

- Clicking an attribute, for example **Discovered OS Class**, adds a **Host: Discovered OS Class** entry to the Query Viewer.
- Clicking a relationship, for example **Software Instance: Software Instances running on this host**, displayed with a following right arrow (→), populates the next pane in the selection tool with the attributes and relationships of the destination node. Click an attribute here to add it to the Query Viewer.
- Clicking a relationship in the second pane populates the third pane in the selection tool with the attributes and relationships of the destination node in the same way as before. Click an attribute here to add it to the Query Constructor.
- Clicking a further relationship populates a fourth pane and scrolls the previous panes to the left, hiding the first pane. You can scroll back by clicking the arrow to the left of the selector panes.

**How traversal selection works**

When you need to traverse to another node kind, click the **Traverse to** link. The traversal selector replaces the three pane selector. From this you can select the node type to traverse to, and select the relationship to use for the traversal. The example below shows a traversal from a host to a Discovery Access node using a failed access relationship.
SI to SI traversals

The following relationships are among those available when specifying a traversal from Software Instance to Software Instance:

- Software Instances observed to be communicating with this Software Instance
- Server Software Instances that this Software Instance is communicating with (Client to Server Comms)
- Client Software Instances that are communicating with this Software Instance (Server to Client Comms)
- Peer Software Instances that are communicating with this Software Instance (Peer to Peer Comms)

The last three are "explicit", in other words built by a discovery for later use. Examples of these are relationships created by patterns that parse application configuration files to find communicating hosts.

The first relationship is a "pseudo-traversal" based on the communicatingSIs function. While the pseudo-traversal might be useful where relationships corresponding to the explicit relationships do not exist, it should be used with caution due to its potential impact on performance. These relationships can be created from the output of commands such as netstat.

The first relationship is only available in the the Traverse to dialogue and not the Filter dialogue. You cannot filter by it because it is not a key expression.

Query Viewer

The Query Viewer provides a map of the query that you are constructing. You can evaluate conditions on an attribute or group of attributes using the following conditions:

- All: True when ALL conditions are true
- Any: True when any of the conditions are true
- None: True when none of the conditions are true

These conditions are selected using a drop-down selector in the container which holds the attribute or attributes of interest. For example, in the following illustration, two attributes have been selected from the first pane and are grouped with an All condition.
In this example, the Discovered OS Class is still being evaluated. A Software Instance has been added by scrolling down the list to Software Instance: Software Instances running on this host when this is clicked, the second pane is populated with the attributes and relationships of Software Instances. The Name attribute has been added to the query viewer by scrolling down the second pane list and clicking Name.

You can also traverse from the host.

To build a query

This section illustrates an example of how to use the Query Builder to determine which hosts in the organization are web servers. The query should find all Windows hosts running Apache or IIS and all UNIX hosts running Apache.

1. From a host list view, click the Customize control under the page title.
2. Click the Query Constructor tab.
3. From the first pane, select Discovered OS Class from the list of host attributes.
   The Discovered OS Class attribute is added to the Query Viewer.
4. Type "Windows" in the text entry field and leave the condition drop-down on contains word.

In this example, you are initially looking for Windows hosts running IIS and need to create an nesting level where this first test can be performed.
5. Click the **Add Condition** icon in the Discovered OS Class row.
   A new container is added which will be used to supply the All, Any, or None condition to be evaluated. Before you add a Software Instance, you must ensure that it is added in the correct place.

6. Click in the area of the container you just added to set the focus on the container.
   When selected, the container is highlighted in yellow.

7. Scroll down the first pane to **Software Instance: Software Instances running on this host**.

8. Click this to populate the second pane with the attributes and relationships of Software Instances.

9. From the second pane, select the **Name** attribute.

10. Type "IIS" in the text entry field and leave the condition drop-down on **contains word**.

11. Click **Apply** to display the results in the list view.

Now that you have a list of Windows hosts that are running IIS, you must find UNIX hosts running Apache.

12. With the root of the Query Viewer highlighted, select **Discovered OS Class** from the list of host attributes.
   The **Discovered OS Class** attribute is added to the Query Viewer outside the section of the query completed above.

13. Type "UNIX" in the text entry field and leave the condition drop-down on **contains word**.

14. Click the **Add Condition** icon in the Discovered OS Class row.
   A new container is added which will be used to supply the All, Any, or None condition to be evaluated.

15. Click the container to select it.

16. Scroll down the first pane to **Software Instance: Software Instances running on this host**.

17. Click the name to populate the second pane with the attributes and relationships of Software Instances.

18. From the second pane, select the **Name** attribute.
19. Type "Apache" in the text entry field and leave the condition drop-down on contains word.
20. Change the root condition drop-down to Any.
21. Click Apply to display the results in the list view.

You know that some of the web servers are on hosts which are members of clusters. To list those clusters, you must traverse to the cluster nodes.

22. Click the Traverse to link.
   The Query Constructor is replaced by a Traverse via Relationship pane.
23. Click Cluster.
   The Traverse via Relationship pane is populated with the available relationships. In this case, the Cluster of which this is a member relationship.
24. Click Apply.
   The List of hosts is replaced with a list of clusters of which hosts in the previous hosts list are members, and the Traverse via Relationship panel is replaced with the Query Constructor pane. You can use the Query Constructor pane to filter the result set further.
25. Click Save to save the query.
   Saved queries can be accessed by clicking Saved Queries on the Reports Tab. The list of saved queries is restricted to queries you have created; queries saved by other users are not displayed.
26. You can run a query again by clicking through to the query and selecting Run Query from the Actions menu.

Exporting data in CSV format
You can save the output from searches or reports in CSV format from the BMC Atrium Discovery UI. The export option is not available in the BMC Atrium Discovery Community Edition.

BMC Atrium Discovery Export
BMC Atrium Discovery Export enables you to publish data from BMC Atrium Discovery directly into relational databases and into all systems that can accept CSV files. BMC Atrium Discovery data is exported by creating an exporter, which is a combination of a mapping set with an adapter configuration. These concepts are explained in Exporting data.

To export data in CSV Format
You can export the results of any search operation or any report in CSV format, enabling you to access the data by using Microsoft Excel or other spreadsheet and database applications.
1. Click **Export as CSV** on the results page of any search or report. You are prompted to open the CSV file directly or to save it.
2. *(Optional)* You can change the default file name, which is **export.csv**. CSV files can be opened and edited with any text editor, spreadsheet, or database program.

**Export API**

BMC Atrium Discovery is provided with an API that enables you to create custom scripts to interrogate the datastore. The export API can export data in XML or a stream of comma-separated text or an empty string.

See [Export APIs](#) for more information about importing and exporting data.

**Using View Object pages**

A View Object page is displayed when you click a linked object from a List page or another View Object page. It displays more detailed information about the selected object, such as the attributes of the object and its relationships with other objects. Relationships to other objects are indicated by a highlighted hyperlink. When you click one of these links, the View Object page of the linked object is displayed. For advanced users wanting to understand the BMC Atrium Discovery model in greater detail, see the following information about the model.

**Note**

Only those attributes and relationships of the object that have been set appear; any that are unknown are not displayed.

**What you can do from a View Object page**

- Show the provenance of the object by clicking **Show Provenance**.
- View the history of the object by selecting **History** from the Actions menu. See [Comparing the history of nodes](#).
- Compare the object to other like-objects by selecting **Compare To...** from the Actions menu. See [Node to Node Comparison](#).
- Destroy the object by selecting **Destroy** from the menu. See [Destroying data](#).
- Display details of a related object by clicking on a hyperlink. The View Object page of the selected object is displayed.
- Display a **Software context** (a view that shows the Object in context of the model - with all related objects).

**BMC Atrium Discovery data is read-only**
BMC strongly recommends that you do not edit any nodes as your version will not be retained by BMC Atrium Discovery. However, if you need to enable the edit facility, you must contact your BMC Atrium Discovery Administrator.

Viewing software context

The software context view provides a quick overview showing how the object(s) (host, software instance, load balancer, mainframe, and so on) connect(s) into the rest of the model.

This topic contains the following information about viewing and managing the software context view:

- Launching the software context view
- Key software context information
- Minimizing and maximizing software context view
- Changing a software context focus
- Zooming and moving items around
- Filtering the software context
- Showing labels
- Changing the software context layout
- Closing software context view

Launching the software context view

The Software Context view is only supported in modern standards-compliant web browsers. It works best in recent versions of Chrome and Firefox, and it is also supported in Internet Explorer version 10 and later. It is not supported in Internet Explorer 9 or earlier.

Software context for objects

On the object details page, a minimized software context preview is shown by default, like in the example below.
Software context for lists

You can open software context view for a custom set of nodes by selecting them in the object list and then using **Software Context** action. By default, the software context for a list of objects is maximized.

The following example shows the software context view of two clusters that were selected manually:
Key software context information

To view the key that explains colors and shapes of the nodes and relationships, as well as navigation basics, click following icon:

Below is an example of the information that is provided:

```
Software Context Information

- Arrows show impact direction
- Click on icons to navigate
- Drag icons to rearrange the diagram
- Drag background to move the whole diagram
- Use scroll wheel to zoom
- Double-click to recenter and zoom to fit

Current Cluster

- Business Application Instance
- Load Balancer Service
- Cluster
- MFPart
- Database
- Mainframe
- Host Container
- Second-order Software Instance
- Host
- Software Component
- Load Balancer Instance
- Software Instance
- Load Balancer Member
- Virtual Machine
- Load Balancer Pool

Hosted Software
- Software Containment
Host Containment
- Database Containment
Communication
- Containment
Observed Communication
- Software Service
Dependency
- Management
```
Minimizing and maximizing software context view

You can toggle the software context window size by switching to a floating pop-up dialog (-) or to a maximized docked panel at the top of the Object Details page (+).

Changing a software context focus

In the software context window the icon that represents the object you are looking at is highlighted. When you hover your mouse over nodes, tooltips are displayed which show the names of the nodes. Click on a node to view the object page of that node. When you do so, the software context window is retained, and the node whose object page you are viewing is highlighted. This provides a quick and simple method of exploring connected nodes that might make up an application, or the virtual machines hosted on a server and so on.

Zooming and moving items around

In both docked (maximized) and floating (minimized) views, use the mouse scroll wheel to zoom in and out, and drag on the background to move around. On a touch device, pinch to zoom. Double-click inside the view to recenter and zoom the display to fit the window. Drag the background to move the whole diagram. Drag individual icons to rearrange the view.

Filtering the software context

In the maximized view, you can filter the information shown as the software context by showing /hiding nodes of some types.

Click on the Show/Hide list, select the node types that you would like to be displayed and deselect those that should be hidden.
Showing labels

The labels are visible only in the maximized view. Tick **Show labels** to show the text label for every node. When labels are hidden, the view is more compact and relationships directions might look clearer. You can see the label hint when you hover over the node.

Changing the software context layout

In the maximized view, you can auto-arrange the nodes and relationships layout using the following templates:

- **Switch wide/tall**— Optimizes the graph for horizontal/vertical view.
- **Force-directed**— Arranges nodes so that the impacting nodes have more central position than the impacted ones.
- **Impact**— Levels the impacted objects in several layers depending on the impact depth.
- **Root at top**— Arranges nodes into a tree view with a root node at the top.

You can also drag nodes manually to tune the layout for the best fit for your specific case.

Closing software context view

You can close software context when you don't need it using (x), and reopen by clicking Software Context.

Searching for data

You can search for information in the BMC Atrium Discovery datastore from the drop-down Search box that is displayed at the top-right of each page. You can run a quick search, basic search, or an advanced search that includes additional options, including allowing you to define the set of objects to be searched and to search for destroyed objects. See [Version Number and Search Bar](#).

- To perform a quick search
- To perform a basic search
- How advanced searching works
- To perform an advanced search
- To search on a regular expression

To perform a quick search

1. In the top search box, enter the keyword or text string you want to search for. To search for a phrase, enclose it in quotes (for example "apache webserver").
2. Hit **Enter** or click the arrow button.

To limit the kinds of nodes found by the search, prefix the search terms with the node kind followed by a colon, for example **host: test.example.com** finds Host nodes matching "test.example.com".

The node kind is case insensitive. You can use any node kind defined in the taxonomy, plus the following abbreviations:

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Node kinds searched</th>
</tr>
</thead>
<tbody>
<tr>
<td>tku</td>
<td>Pattern, PatternModule and KnowledgeUpload</td>
</tr>
<tr>
<td>si</td>
<td>SoftwareInstance</td>
</tr>
<tr>
<td>process</td>
<td>DiscoveredProcess</td>
</tr>
<tr>
<td>pattern</td>
<td>Pattern and PatternModule</td>
</tr>
<tr>
<td>da</td>
<td>DiscoveryAccess</td>
</tr>
<tr>
<td>bai</td>
<td>BusinessApplicationInstance</td>
</tr>
<tr>
<td>application</td>
<td>BusinessApplicationInstance</td>
</tr>
<tr>
<td>app</td>
<td>BusinessApplicationInstance</td>
</tr>
</tbody>
</table>

**To perform a basic search**

A basic search enables you to search for keywords or text strings occurring anywhere in the BMC Atrium Discovery datastore.

1. In the top search box, enter the keyword or text string you want to search for.
2. Under Section, choose the section of the datastore (Administration, Applications, Discovery, Infrastructure, or Change) that you want to search.
   The Any option searches all modules. See **Notes on Searching** for a list of the objects in each module.
3. Under Match, choose the type of search you want to run.
   - **Exact Match** — Requires an exact match for an entire attribute. This search is case-sensitive.
   - **Word Match** — Searches for entire words (separated by spaces or punctuation) and does not find partial words or substrings. For example, if you search for "Windows", the system will return Windows XP, Windows 2000, and so on, but if you are searching for a person called Robertson, the search does not find details if you enter "Robert". This search is not case-sensitive.
4. The Show Data Completeness check box indicates whether or not the Data Completeness indicator will be shown for each returned object. Select this check box if you need to view the data quality for all items. The default setting allows for faster searching when large numbers of objects are involved, as the system will not need to calculate data completeness values.
5. Click **Search**.
5. If objects of more than one kind are found, a page is displayed listing the number of objects of each kind found.
   Click any entry in this list to display the list of objects of this kind.
6. If objects of only one kind are found, the matching ones are listed immediately. The list shows Summary attributes only, and it is possible that the text you searched for does not appear in these attributes.

6. Click any entry to display details of the object.

**Tips on searching**

- If your search returns too many matches, you can refine it by running a further search on the items.
- You can export the results of a search in CSV format. See [Exporting data in CSV format](#).

**How advanced searching works**

In an advanced search, you can search a defined set of objects for keywords or text strings, and you can opt to include destroyed objects as well as restrict your search to your own objects. You can also match against a regular expression.

If you are familiar with the Search Service and its query language you can use the Advanced Search page to enter a search query. To access the Advanced Search page, from the drop-down Search box, click Advanced Search. For a detailed description of the Search Service and query language, see the [Using the Search and Reporting service](#). The following check boxes are provided in each advanced search section:

<table>
<thead>
<tr>
<th>Section</th>
<th>Check box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applications</td>
<td>Application Instances</td>
</tr>
<tr>
<td></td>
<td>Groups</td>
</tr>
<tr>
<td></td>
<td>Pattern Generation Information</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Clusters</td>
</tr>
<tr>
<td></td>
<td>Collections</td>
</tr>
<tr>
<td></td>
<td>Coupling Facilities</td>
</tr>
<tr>
<td></td>
<td>Database Details</td>
</tr>
<tr>
<td></td>
<td>Details</td>
</tr>
<tr>
<td></td>
<td>Disk Drives</td>
</tr>
<tr>
<td></td>
<td>Fibre Channel HBAs</td>
</tr>
<tr>
<td></td>
<td>File Systems</td>
</tr>
<tr>
<td></td>
<td>Files</td>
</tr>
<tr>
<td></td>
<td>Generic Elements</td>
</tr>
<tr>
<td></td>
<td>Host Containers</td>
</tr>
<tr>
<td></td>
<td>Hosts</td>
</tr>
<tr>
<td></td>
<td>IP Addresses</td>
</tr>
<tr>
<td></td>
<td>Load Balancer Groups</td>
</tr>
<tr>
<td></td>
<td>Load Balancer Instances</td>
</tr>
<tr>
<td></td>
<td>Mainframes</td>
</tr>
<tr>
<td></td>
<td>Network Devices</td>
</tr>
<tr>
<td></td>
<td>Network Interfaces</td>
</tr>
<tr>
<td></td>
<td>Packages</td>
</tr>
<tr>
<td></td>
<td>Patches</td>
</tr>
<tr>
<td></td>
<td>Printers</td>
</tr>
<tr>
<td></td>
<td>Runtime Environments</td>
</tr>
<tr>
<td></td>
<td>SNMP Managed Devices</td>
</tr>
<tr>
<td></td>
<td>Software Components</td>
</tr>
<tr>
<td></td>
<td>Software</td>
</tr>
<tr>
<td></td>
<td>Instances</td>
</tr>
<tr>
<td></td>
<td>Storage</td>
</tr>
<tr>
<td></td>
<td>Storage Collections</td>
</tr>
<tr>
<td></td>
<td>Storage Devices</td>
</tr>
<tr>
<td></td>
<td>Storage Pools</td>
</tr>
<tr>
<td>Section</td>
<td>Check box</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Load Balancer Members</td>
<td>Storage System Groups</td>
</tr>
<tr>
<td>Load Balancer Pools</td>
<td>Storage Systems</td>
</tr>
<tr>
<td>Load Balancer Services</td>
<td>Storage Volumes</td>
</tr>
<tr>
<td>MF Parts</td>
<td>Subnets</td>
</tr>
<tr>
<td>Discovery</td>
<td>Discovered Software</td>
</tr>
<tr>
<td></td>
<td>Discovered Software Lists</td>
</tr>
<tr>
<td></td>
<td>Discovered Storage Subsystem Lists</td>
</tr>
<tr>
<td></td>
<td>Discovered Storage Subsystems</td>
</tr>
<tr>
<td></td>
<td>Discovered Sysplex Lists</td>
</tr>
<tr>
<td></td>
<td>Discovered Sysplexes</td>
</tr>
<tr>
<td></td>
<td>Discovered Tape Drive Lists</td>
</tr>
<tr>
<td></td>
<td>Discovered Tape Drives</td>
</tr>
<tr>
<td></td>
<td>Discovered Transaction Lists</td>
</tr>
<tr>
<td></td>
<td>Discovered Transactions</td>
</tr>
<tr>
<td></td>
<td>Discovered Virtual Machines</td>
</tr>
<tr>
<td></td>
<td>Discovered WBEM Associations</td>
</tr>
<tr>
<td></td>
<td>Discovered WBEM Associates Results</td>
</tr>
<tr>
<td></td>
<td>Discovered WBEM Instances</td>
</tr>
<tr>
<td></td>
<td>Discovered WBEM Queries</td>
</tr>
<tr>
<td></td>
<td>Discovered WBEM Query Results</td>
</tr>
<tr>
<td></td>
<td>Discovered WMI Queries</td>
</tr>
<tr>
<td></td>
<td>Discovered WMI Query Results</td>
</tr>
<tr>
<td></td>
<td>Discovery Accesses</td>
</tr>
<tr>
<td></td>
<td>Discovery Conditions</td>
</tr>
<tr>
<td></td>
<td>Discovery Runs</td>
</tr>
<tr>
<td></td>
<td>ECA Errors</td>
</tr>
<tr>
<td></td>
<td>FQDN Lists</td>
</tr>
<tr>
<td></td>
<td>File System Lists</td>
</tr>
<tr>
<td></td>
<td>HBA Info Lists</td>
</tr>
<tr>
<td></td>
<td>Hardware Reference Data</td>
</tr>
<tr>
<td></td>
<td>Host Infos</td>
</tr>
<tr>
<td></td>
<td>Integration Points</td>
</tr>
<tr>
<td></td>
<td>Integration Results</td>
</tr>
<tr>
<td></td>
<td>Knowledge Uploads</td>
</tr>
<tr>
<td></td>
<td>Listening Ports</td>
</tr>
<tr>
<td></td>
<td>Network Connection Lists</td>
</tr>
<tr>
<td></td>
<td>Network Connections</td>
</tr>
<tr>
<td></td>
<td>Pattern Definitions</td>
</tr>
<tr>
<td></td>
<td>Pattern Definitions Functions</td>
</tr>
<tr>
<td></td>
<td>Pattern Executions</td>
</tr>
<tr>
<td></td>
<td>Pattern Modules</td>
</tr>
<tr>
<td></td>
<td>Patterns</td>
</tr>
<tr>
<td></td>
<td>Process Lists</td>
</tr>
<tr>
<td></td>
<td>Provider Accesses</td>
</tr>
<tr>
<td></td>
<td>Registry Listings</td>
</tr>
<tr>
<td></td>
<td>SQL Result Rows</td>
</tr>
<tr>
<td></td>
<td>Script Failures</td>
</tr>
<tr>
<td></td>
<td>Service Lists</td>
</tr>
<tr>
<td></td>
<td>Session Results</td>
</tr>
<tr>
<td></td>
<td>Virtual Machine Lists</td>
</tr>
</tbody>
</table>

Administration
To perform an advanced search

1. In the drop-down Search box, click Advanced Search. The Advanced Search box is displayed in the main frame.

2. If you are familiar with the Search Service and its query language you can use the Search Query page to enter a search query. To access the Search Query page, click the Generic Search Query link.

For a detailed description of the Search Service and query language, see the Using the Search and Reporting service.

3. Under Keywords, enter the keyword or text string you want to search for. (To run a Regular expression search you must enter a valid regular expression.)

4. In the Select match type option, choose the type of matching to be performed:
   - Exact Match - Requires an exact match for an entire attribute. This search is case-sensitive.
   - Word Match - Searches for entire words (separated by spaces or punctuation) and does not find partial words or substrings. For example if you search for "Windows", the system will return Windows XP, Windows 2000, and so forth, but if you are searching for a person called Robertson, the search does not find details if you enter "Robert". This search is not case-sensitive.
   - Partial Match - Finds all occurrences of the string anywhere in the object's attributes. This search is not case-sensitive.
   - Regular Expression Match - Interprets the input keyword as a regular expression and matches against it. (See Regular Expression Searching.)

5. Select the Include Destroyed check box if you want the search to include objects that have been destroyed. These will otherwise not be included in the search.

6. Select the Show DQ check box if you need to view the Data Quality for all items. The default setting allows for faster searching when large numbers of objects are involved, as the system will not need to calculate data quality values.

7. Select check boxes to indicate the kinds of object in each module (Applications, Infrastructure, Discovery, and Administration) that you want to search. You can select any number. In each module section you can click All to choose all kinds in the module or None to deselect all kinds of objects in the module.

8. Click Search.
   - If objects of one kind are found, the matching ones are listed. The list shows Summary attributes only, and it is possible that the text you searched for does not appear in these attributes.

<table>
<thead>
<tr>
<th>Section</th>
<th>Check box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attachment Categories</td>
<td>Locations</td>
</tr>
<tr>
<td>Attachments</td>
<td>Organizational Units</td>
</tr>
<tr>
<td>Charts</td>
<td>People</td>
</tr>
<tr>
<td>Families</td>
<td>Recovery Times</td>
</tr>
<tr>
<td>Lifecycle Status</td>
<td>Reports</td>
</tr>
</tbody>
</table>
• If objects of more than one kind are found, a page is displayed listing the number of objects of each kind found. Click an entry in this list to display the list of objects of this kind.

• If the object that you were searching for was not found, click Search Again to return to the Advanced Search screen.

9. To display the View Object page of the objects found, click the object name.

Tips on searching

• The returned list displays summary attributes of each object only. Click an item in the list to access the View Object page which displays all of an object’s relationships and attributes.

• If your search returns too many matches, you can refine it by running a further search on the items. If the matches include destroyed items, an Include destroyed items check box is shown. Enter an additional keyword and a matching type and click Refine Search.

• You can export the results of a search in CSV format. See Exporting data in CSV format.

To search on a regular expression

The advanced search option in BMC Atrium Discovery enables you to search by matching against a regular expression. A regular expression, or regex, is a pattern that can match various text strings. For example, A[0-9]+ matches any string that consists of the letter A followed by one or more digits.

Regular expressions have a defined syntax which enables you to define complex matching patterns. BMC Atrium Discovery uses the Python implementation; for full syntax and details of use, consult the Python documentation, see https://docs.python.org/2/library/re.html for more information.

Below are just a few of the matching characters that you can use when constructing regular expressions.

An ordinary character, or a sequence of characters, matches that character or string.

<table>
<thead>
<tr>
<th>Character</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>A dot matches any single character.</td>
</tr>
<tr>
<td>^</td>
<td>A caret matches characters at the start of the string.</td>
</tr>
<tr>
<td>$</td>
<td>A dollar sign matches characters at the end of the string.</td>
</tr>
<tr>
<td>*</td>
<td>An asterisk matches 0 or more repetitions of the preceding regex. For example, ab* will match &quot;a&quot;, &quot;ab&quot;, or &quot;a&quot; followed by any number of &quot;b&quot;s.</td>
</tr>
</tbody>
</table>
### Comparing the history of nodes

This facility enables you to compare any like-nodes in BMC Atrium Discovery to identify and monitor any differences over time. For example, you can display two selected nodes side by side to view where the differences are or which parts are unique to that particular version. The various results are color-coded to make them easier to identify.

The historical comparison facility applies to any kind of node (object) in the system. However, the examples below illustrate the various views and comparison options which are available for a selection of host nodes in BMC Atrium Discovery. For more information about attributes and relationships for the following examples, see Viewing a host.

The main comparison capabilities across all nodes in BMC Atrium Discovery are:

- Historical comparison of an entire node
- Node to node comparison

#### Historical comparison of an entire node

You can view details of the history of an entire node and display details of how the node has changed over time, including the changes made, the date and time of the change, the name of the user who made the change and so forth.

**To view a node for historical comparison**

1. Display the View Object page of the relevant node (for example, a host).
   ![View Object Page]

2. From the Actions drop down, select **History**.

3. The Historical Comparison page is displayed for the node you have selected, as shown in the following screen.
4. The default view for the Historical Comparison page initially shows only the attributes that have been changed on this node since it was first created.

⚠️ Grayed-out cells indicate that there is no attribute present.

5. Click the **Show All Attributes** link if you want to display all of the attributes for this particular node.
6. Click the **Show Differences Only** link to return to the default view.

**Historical Comparison page**

The Historical Comparison page is divided into the following columns.

- **Attribute Column** — Displays the attributes of this node.
- **Center Column** — Displays a version of the node you have selected for historical comparison. The default view for this column shows the last time that something was amended for this node, that is, version 3 of 4 for this example above.
- **Right Column** — Displays a particular version of the node you have selected for historical comparison. The default view for this column shows the latest version of this node, that is, version 4 of 4 for this example above.

The default view, in the first header row of the right column, shows the most recent state of the node on 18/11/2013 at 10:34 and that it is the forth version of this node (4 of 4). The fact that version 4 of this node is being displayed in the right column is also indicated in the second header row where the number 4 is not underlined.

The previous change (3 of 4) was made to this node on 18/11/2013 at 10:27.

This screen illustrates a change to a node from version 3 to 4.

The second header row enables you to independently change the versions of the node to display different historical comparisons. For example, you might want to compare version 2 (2 of 4) with version 3 (3 of 4) to identify exactly what changed during that period of time.

The blue direction arrows in this row enable you to view earlier or later versions of this node and are configured as follows:

- **Arrows pointing left** — Click once to display the next earlier version.
- **Arrows pointing right** — Click once to display the next later version.
Alternatively, you can click the version number directly in each column. The historical comparison between the two versions that you select is displayed automatically.

Three types of information changes are color-coded and recorded on this page:

Green — The attribute is unique to the node version in the left column.
Yellow — The attribute differs across node versions selected.
Red — The attribute is unique to the node version in the right column.

Attributes for different nodes in BMC Atrium Discovery might vary, however, the first attribute row is Change Made By and is always present. This identifies the user that made the change. Remaining attributes for each node are listed in Taxonomy defined order.

You can also view a separate list of data for certain attributes which shows unique and common items. For this example you can click the links for each version for Patches, Packages and Network Interfaces to view this information. For example, you can click the link in either node version column for the Patches attribute.

Click the link in either node version column again to return to the Communication Links attribute summary view.

To change the history compression threshold

When using the History view described above what you are in fact seeing is a processed version of the underlying raw data. The historical comparison pages 'compress' the raw history into changes. Entries are grouped by user and a sliding time threshold. This time threshold is necessary to enable you to view a number of separate changes made by Reasoning as a single change. You can configure this time threshold from the Administration->Appliance->Miscellaneous Settings page.
To switch from raw to comparative history views

You choose to view historical changes in their raw form, without the "compression" described previously. To switch between the two modes of viewing history, click the Switch to raw view link at the top-right of the page.

The view changes to one where each individual change is output per line.

To change back to the compressed view, click the **Switch to comparative view** link.

You can change which type of view you would like to have appear by default by changing your preferences.

Node to node comparison

You can also perform node to node comparisons for any kind of node. However, you can only compare like with like (for example, host to host). The comparisons are completely arbitrary and they do not need to have a relationship, as long as they are like for like nodes.

The following example relates to a host-to-host comparison.

To select nodes for node to node comparison

1. Display the View Object page of the relevant host. The BMC Atrium Discovery Host page is displayed.
2. From the Actions menu, select **Compare To**.
3. The Search for Hosts to Compare page is displayed for the node you have initially selected.

   You can also change the details of this first host to search for another host.
4. Click the **Change** link.
5. In the Search for the first Host to Compare pane, type a keyword for the name or the IP address of the first host you want to compare.
6. Select the type of Match from the drop-down menu and click **Search**.
7. A list of results is displayed.
8. Select a host to compare.
Note

To return to the default details for the first host at any time, click **Cancel**.

9. In the Search for the second Host to Compare pane, type a keyword for the name or the IP address of the second host you want to compare.
10. Select the type of Match from the drop-down menu and click **Search**.
    A list of results is displayed.
11. Select a second host to compare against the first host and click the name of the host.
    The following screen illustrates what is displayed.

This screen illustrates the results of a host-to-host comparison.

12. After you have selected the two hosts that you want to compare, select **Compare** from the Actions menu.

Note

The width of each displayed column is dependent on the data content of each host.

The node to node comparison page is divided into the following columns.

- **Attribute Column** — Displays the attributes of this node.
- **Center Column** — Displays a version of the node you have selected for historical comparison.
- **Right Column** — Displays a particular version of the node you have selected for historical comparison.

The default view, in the first header row of the left and right columns, displays the most recent state of each separate node.

The second header row enables you to independently change the versions of each separate node to enable you to display different historical comparisons. For example, you might want to compare version 3 of de-32-R8-2 with version 27 of lon-vm-nightrun03.

The blue direction arrows in this row enable you to view earlier or later versions of this node and are configured as follows:
- **Arrows pointing left** — Click once to display the next earlier version.
- **Arrows pointing right** — Click once to display the next later version.

Alternatively, you can click the version number directly in each column. The historical comparison between the two versions that you select is displayed automatically.

One type of information change is color-coded in Yellow and recorded on the page, indicating that information differs across selected separate nodes, attributes or attribute items.

You can view a separate list of data for certain attributes in each node which shows unique and common items particular to that node. For this example, you can click the links to show detailed information for each separate node for the following attributes:

- Software Instances
- Runtime Environments
- Containing VM
- Packages

For example, you can click the link in either node column for the Packages attribute.

![Comparison of de-32-R8-2 with lon-vm-nightrun03](image)

This screen illustrates that node _de-32-R8-2_ has 4 packages that are unique and 307 packages that it has in common with node _lon-vm-nightrun03_.

This example illustrates that node _de-32-R8-2_ has 4 packages that are unique and 307 packages that it has in common with node _lon-vm-nightrun03_. It also shows that node _lon-vm-nightrun03_ has 4 packages which are unique.

For any of the listed patches on either node, you can click the radio button next to it to select that particular patch. This facility enables you to compare individual patches (or other attributes) across the two nodes that you have selected for comparison.

To compare individual attributes:

1. Click the radio button next to an individual patch for each of the two nodes.
2. Click **Compare**.
3. Click **Compare** again to return to the previous page.
4. Click the link to return to the summary view.

To change the nodes that you are comparing at any time, click the Change Comparison link at the bottom of the page.

Using this facility, you can compare any like-nodes in BMC Atrium Discovery with each other, their individual attributes and individual items in selected attributes over any period of time.

**Viewing dependency visualizations**

BMC Atrium Discovery enables you to view object dependencies in graphical format, known as dependency visualizations. You can visualize relationships that are not tree structures.

- To view a dependency visualization
- To view multiple nodes
- To observe host communication
- Working with the dependency visualization controls and legend

**To view a dependency visualization**

When viewing either a single node or multiple node groups, a Visualization menu is shown.

1. Click the Visualizations menu to display the menu.

![Visualization menu]

*This screen illustrates the Visualization menu.*

2. Click the dependency visualization type you want to view (for example, the Running Software visualization).
This screen illustrates a visualization that displays the infrastructure for a host. There are many different types of visualization available for viewing each object. If you click any of the types here, a pictorial representation of the object dependencies is displayed. See Types of visualization for more details. It is also possible to create custom made Dependency Visualizations on your system using a configuration file. See Configuring dependency visualizations for more information.

⚠️ Visualizations and Navigation

When viewing dependency visualizations you should avoid using the Forward and Back buttons on your browser. Doing so might cause problems with the view (for example, the zoom function might be erratic, and navigation buttons in the display applet might not work).

To view multiple nodes

When viewing a result set containing multiple nodes, a visualization is displayed with multiple root nodes; one for each node in the result set. Opening a visualization on many nodes at once might be slow. For this reason, you are asked for confirmation if you attempt to do this on more than 20 nodes.

To observe host communication

To see details of why two hosts are communicating, right-click the edge between them, and choose the only menu option, Show Communication Details.

A report of all directly discovered network connections that contributed to the observed communication link between the hosts is displayed.
This screen illustrates a report of all directly discovered network connections contributing to an observed communication link between hosts.

**Working with the dependency visualization controls and legend**

Several controls and a legend are provided to enable you to work with dependency visualizations:

- Zoom and pan controls enable you to zoom in and out, and pan around the visualization.
- Layout controls enable you to choose a layout.
- Printing and export controls enable you to print or export the visualization image.
- The legend shows an annotated legend describing the node types shown in the visualization.

In addition to the control panels, further options are available through a menu which is displayed by right clicking a node:

- **View Details** — Shows the view object page for the object.
- **Re-Center** — Centers and makes the selected node the subject of the visualization.
- **Manual Groups** — Opens the Manual Groups dialog, allowing you to add and remove selected nodes from manual groups.
- **Hide** — Hides the object.
- **Unhide Neighbors** — Reveals any hidden neighbor objects.
- **Open** — Displays the Hosts in that group, This is only available on an Automatic Group node.

An observed communication link also has a popup menu which is displayed by right clicking it. This allow you to see the Communication Summary between two nodes that are communicating.

**Using the zoom and pan controls**

The diagram is initially centered in the page. If a large number of dependencies are shown you might have to use the zoom facility or the pan up, down, left and right arrows to view the entire visualization. The zoom and pan controls are displayed in the top-left corner of the screen. This panel includes zoom in and out functions and a thumbnail version of the dependency visualization selected.
Using the layout controls

The layout controls provide the ability to display the dependency visualization in four layout styles: Circular, Hierarchical, Orthogonal and Symmetric. The layout shown in the example at the top of this page is the Hierarchical layout.

Nodes can be rearranged by clicking them and dragging them to a new location. Multiple nodes can be selected and dragged together. The icon on the far right of the layout row, separated by a |, is the Incremental Layout tool. This re-displays the visualization to fit on screen, respecting where you have re-arranged your nodes.

Each visualization also resizes if the window is resized.

Using the printing and export controls

The Printing & Export Controls area in the side panel of the screen for each dependency visualization provides a drop-down dialog where you can print or save the current whole image view at any time.

The available paper size options are A0, A1, A2, A3, A4, Letter, and ANSI B - F. Once you have selected your paper size, click Print/Export and a new screen is displayed from where you can select your printer.

⚠️ Select Landscape when Printing

Whichever page size you select you must select the Landscape option from the print dialog.

The following method can be used to export your visualization:

1. Click Print/Export and a new screen is displayed.
2. Right-click the image and choose the Save Picture As link.

⚠️ Visualization size limit

There is a visualization size limit of 5631x4439 pixels. Larger visualizations could consume excessive system resources and cause instability.

Using the legend

There is a list of the icons and their meanings in the side panel. The list includes everything that appears in the dependency visualization. The legend also displays all the Manual Group labels that are being displayed in the visualization.
This screen illustrates the visualization legend.

The node that is the root of the visualization, that is, the one from which the visualization was launched, or on which it was re-centered, is highlighted with an orange box. For an example of a multiple root node visualization, see Multiple nodes.

**Types of visualization**

Dependency visualizations are viewed from a dynamic list of links depending on the node kind being viewed. There are a number of different default types of dependency visualization available for viewing different objects. If you click any of the types listed below, a diagrammatic representation of the object dependencies is displayed. The presence of links is determined by the node kind being viewed. For example, clustering visualizations are available to view on cluster nodes, host nodes and software instance nodes.

The following links are provided in the Visualization drop-down dialog:

- Application Dependencies - Application View
- Application Dependencies - Software View
- Application Functional Component
- Automatic Grouping Detail
- Clustering
- Containment
- Coupling Facility Infrastructure
- Database Detail
- File System Dependencies - Detail
- File System Dependencies
- Host Containment
- Host Containment with Software
- Inferred Software Communication
- Inferred Software Dependency
About Atrium Discovery

- Infrastructure
- Management
- Network Topology - Neighbor
- Observed Communication - Host View
- Observed Communication - Process View
- Observed Communication - Software View
- Software Running on Host
- Virtualization
- Provenance
- Running Software
- Software Structure
- Network Device Services
- Load Balancing

Application Dependencies - Application View

Displays Business Application Instance (BAI) dependencies, from a BAI point of view, explicitly built communication and dependency relationships between BAIs, as well as communication dependencies observed by looking at directly discovered network connections. You can also show hosts and switches, and groups hosts into clusters (shown as boxes) if appropriate. The BAI on which the visualization is centred, has its Software Instances (SIs) shown.

These Visualizations are available on the following nodes:

- Business Application Instances
- Software Instances
- Hosts
- Clusters
- Host Containers
- Switches

This screen illustrates an example of an application dependencies - application view visualization.
Application Dependencies - Software View

Displays BAI dependencies, from an SI point of view, explicitly built communication and dependency relationships between SIs as well as communication dependencies observed by looking at directly discovered network connections. Also shows hosts and switches, and groups hosts into clusters (shown as boxes) if appropriate. This type only shows SIs that are part of a BAI. These Visualizations are available on the following nodes:

- Business Application Instances
- Software Instances
- Hosts
- Clusters
- Host Containers

This screen illustrates an example of an application dependencies - software view visualization.

Application Functional Component

Shows the functional component structure of an application model. To see the classic software structure use #Software Structure

These Visualizations are available on the following nodes:

- Business Application Instances
- Functional Components
- Software Instances
- Software Components
- Database Details
This screen illustrates an example of an application functional component dependency visualization.

**Automatic Grouping Detail**
Displays the automatic group the host or hosts belong to. These Visualizations are available on the following nodes:

- Hosts

**Clustering**
Displays Clusters and the software services that manage them. Clustering Visualizations are available on the following nodes:

- Software Instances
- Hosts
- Clusters

This screen illustrates an example of a clustering dependency visualization from the point of view of a cluster.

**Containment**
Displays the relationship between network devices. This type of visualization is available only on the following node:

- Network
This screen illustrates an example of a containment visualization.

**Coupling Facility Infrastructure**

This visualization displays an infrastructure view only related to the CouplingFacility. This visualization is available only on following node:

- MF Part List

This screen illustrates an example of coupling facility infrastructure dependency visualization.

**Database Detail**

Displays a database's schemas and tables. This assumes that the SI is a database, the extended database patterns are loaded, and we had credentials to log into the database. These visualizations are available on the following nodes:

- Software Instance
- DatabaseDetail

The example below shows this type of visualization.
This screen illustrates an example of a Database Detail visualization.

This screen illustrates an example of database detail dependency visualization.

File System Dependencies - Detail

File System Dependencies- detail focuses on the detail information about file systems. It tries to represent the file systems that are discovered by BMC Atrium Discovery with further intricacies. This visualization is applied to following nodes:

- Hosts
- File Systems

This screen illustrates an example of file system dependencies - detail visualization.

File System Dependencies

Displays file systems included in a node/object, mostly a Host. It displays an overview of the local file systems exported filesystems and dependencies from remote filesystems and their hosts. The Storage TKU displays dependencies related to local Disks and Storage volume mounts. For Virtual Disks or Volume a still deeper approach is utilized. It is available on the same nodes as File System Dependencies- Detail like:

- Hosts
- File Systems
This screen illustrates an example of file system dependencies visualization.

**Host Containment**

Displays Host containers and their contents. This type of dependency visualization is specifically targeted at those host container nodes which are used to represent Sun E10K and E15K machines as well as Solaris Zones.

Host Containment Visualizations are available on the following nodes:

- Hosts
- Host Containers

This screen illustrates an example of a host containment dependency visualization.

This screen illustrates an example of host containment dependency visualization.

**Host Containment with Software**

Displays Host containers, their contents and any running software. This type of dependency visualization is specifically targeted at those host container nodes which are used to represent Sun E10K and E15K machines as well as Solaris Zones.

Host Containment with Software Visualizations are available on the following nodes:

- Hosts
- Host Container
This screen illustrates an example of a host containment with software dependency visualization.

Inferred Software Communication
This type of dependency visualization displays the communication relationships built by patterns. These relationships are explicitly constructed when it is known that two SIs are communicating. Inferred Software Communication Visualizations are available on the following nodes:

- Business Application Instances
- Software Instances
- Hosts

This screen illustrates an example of an inferred software communication dependency visualization.

Inferred Software Dependency
This type of dependency visualization displays the generic dependency relationship built by patterns. These relationships are explicitly constructed when it is known that there is some dependency between two SIs. Inferred Software Dependency Visualizations are available on the following nodes:
- Business Application Instances
- Software Instances
- Hosts

This screen illustrates an example of an inferred software dependency visualization.

**Infrastructure**
Displays the relationships between hosts, switches, and subnets. Infrastructure Visualizations are available on the following nodes:

- Hosts
- Switches
- Subnets

This screen illustrates an example of an infrastructure dependency visualization.

**Management**
This type of dependency visualization shows you the management in a software instance. This visualization is applicable only to the following node:

- Software Instance
This screen illustrates an example of a management dependency visualization.

Network Topology - Neighbor
Displays dependencies between hosts and network devices. This visualization is available on the following nodes:

- Hosts
- Network Device

This screen illustrates an example of a network topology - neighbor dependency visualization.

Observed Communication - Host View
Displays observed communication view showing only hosts. These Visualizations are available on the following node:

- Hosts

This screen illustrates an example of an observed communication dependency - host view visualization.
This screen illustrates an example of an observed communication dependency - host view visualization.

**Observed Communication - Process View**
Displays Hosts shown as boxes with communicating processes inside them and shows host-host, process-host, host-process, and process-process communication information, as observed from network connections.

These Visualizations are available on the following nodes:

- Hosts
- DiscoveredProcesses

This screen illustrates an example of an observed communication dependency - process view visualization.

**Observed Communication - Software View**
Displays Hosts shown as boxes with communicating SIs inside them and shows host-host, SI-host, host-SI, and SI-SI communication information, as observed from network connections.

These Visualizations are available on the following nodes:

- Software Instances
- Hosts
This screen illustrates an example of an observed communication dependency - software view visualization.

This screen illustrates an example of an observed communication dependency - software view visualization.

**Software Running on Host**

This type of dependency visualization displays the software dependency relationships explicitly built by patterns. SIs and BAIs can have a dependency relationship with themselves. Includes Runtime Environments and Software Components in SIs.

Software Dependency Visualizations are available on the following nodes:

- Hosts
- MFParts

This screen illustrates an example of a software dependency visualization.

This screen illustrates an example of a software running on host view visualization.

**Virtualization**

This type of visualization is designed to display virtual machines, for example, VMware. Virtualization Visualizations are available on the following nodes:

- Software Instances
- Hosts
This screen illustrates an example of a virtualization dependency visualization.

Provenance

This type of dependency visualization shows you the provenance, that is, the calculated origin of the data. Provenance dependency visualizations are available for all discovered data as well as for SIs, BAIs and patterns. Provenance Visualizations are available on the following nodes and discovered data:

- Software Instances
- Business Application Instances
- Patterns
- Packages
- Discovered Processes
- Discovered Network Connections
- Discovered Listening Ports
- Discovered Packages
- Discovered Files
- Discovered Command Results
- Discovered Registry Values
- Discovered WMIs

This screen illustrates an example of a provenance dependency visualization.

Running Software

This type of dependency visualization displays the software dependency relationships explicitly built by patterns. SIs and BAIs can have a dependency relationship with themselves. Includes Runtime Environments and Software Components in SIs.

Running Software visualizations are available on the following nodes:
This screen illustrates an example of a software running dependency visualization.

**Software Structure**

Displays the software structure, in terms of how BAIs and higher order SIs are composed. Software Structure Visualizations are available on the following nodes:

- Business Application Instances — including Software Components, and BAIs on MFParts
- Software Instances
- Software Component

This screen illustrates an example of a software structure dependency visualization.

**Network Device Services**

Displays the model of the services network device supports, including the load balancing nodes and relationships.

This Visualization is available on the following nodes:

- Network Device
This screen illustrates an example of a network device services dependency visualization.

**Load Balancing**

Displays Load Balancing dependencies: a device with load balancing enabled, pools configured on a load balancer with all pool members and the services linked between network devices and services and

This Visualization is available on the following nodes:

- Load Balancer Group
- Load Balancer Instance
- Load Balancer Pool
- Load Balancer Member
- Load Balancer Service

This screen illustrates an example of a load balancing dependency visualization.

**Managing attachments**

You can associate a file such as a document, a spreadsheet, or a diagram with certain node types in the BMC Atrium Discovery datastore.

One or more files can be attached to a single object, and copies of a single file can be attached to multiple objects. BMC Atrium Discovery interprets the file type based on the file extension. Make sure that the attachment file extensions are valid. You can view and update a file at any time. To open and view attached files, you must have the appropriate applications.

In a cluster, you can upload an attachment to any machine and it is automatically distributed and a copy is stored on each machine. If you delete an attachment from any machine in the cluster, it is automatically deleted from all machines in the cluster.
The files that you attach are stored as data objects in the BMC Atrium Discovery datastore. No attachment categories are initially defined. Attachments are assigned to a particular category. You can associate a file with certain node types in order to provide additional information. Files can be of any type and are not validated in any way by the BMC Atrium Discovery system. Attached files are uploaded to the BMC Atrium Discovery datastore so that they can be accessed easily for disaster recovery purposes.

To attach a file to an object
You can add attachments to the following node types:

- Host
- Network device
- Printer
- SNMP managed devices
- MFPart
- Software instance
- Business application instance
- Runtime environment

Attaching files to nodes
Node attachment is disabled by default. Attaching files to nodes involves the following steps in the given order:

1. **Enabling node attachment**: You must enable node attachment. To enable node attachment, you must be a user with system group permissions.
2. **Creating attachment categories**: You must create attachment categories from the UI prior to attaching files to nodes. To create attachment categories, you must be a user with admin or system group permissions.
3. **Attaching files to objects**: After creating the attachment categories, you can attach files to nodes from the UI. To attach files to nodes, you must be a user with admin or system group permissions.

Managing node attachments
You can manage the node attachment feature from the appliance's UI.

To enable node attachment

1. Click **Administration > Miscellaneous Settings**.
   Corresponding to Allow attachments to be added to nodes, the value under Current value displays whether node attachment is enabled or not. By default, the value is **No**.
2. To enable node attachment, select **Yes** for Allow attachments to be added to nodes.
3. Click **Apply**.
When you enable node attachment, the value under Current value changes to Yes. If you want to disable node attachment, select No for Allow attachments to be added to nodes and click Apply.

Creating attachment categories
Before you can attach files to nodes, you must create the attachment categories. To learn how to create attachment categories, see Custom Categories.

Managing files attached to objects
From the BMC Atrium Discovery UI, you can manage files attached to objects in the following ways:

- Attach files to nodes
- View files attached to nodes
- Remove files attached to objects

Attaching files to objects
To attach a file to an object, perform the following steps:

1. Display the View Object page of the object that you want to attach the file to.
2. In the Dynamic Toolbox, click the Attachments icon.
3. Click Manage Attachments.
   The Attachments page lists the attached files in each of the available categories.
4. Under the appropriate category, enter the full path of the file in File to Upload or click Browse to select a file.
5. To upload the attachment to the BMC Atrium Discovery system, click Apply.

To view an attached file
After you have attached a file to an object, you can view it directly from the BMC Atrium Discovery user interface, or download it to your system. In both cases, the file remains on the BMC Atrium Discovery system.

1. Display the View Object of the object that has an attachment.
2. In the Dynamic Toolbox, click the Attachments icon to show the drop-down dialog.
   Any files already attached are listed on the drop-down.
3. To check the contents of the file, click the file name.
   The file is opened using an appropriate application if possible.
4. To make a local copy of the file, right-click the name and select Save Target As.

To remove an attached file
You can remove a selected attachment. This removes the instance of that file associated with the selected object. Other copies or local instances of the file are unaffected.

1. Display the View Object of the object that has an attachment.
2. In the Dynamic Toolbox, click the Attachments icon to show the drop-down dialog.
3. Click **Manage Attachments**.
   The Attachments page lists the available categories and any existing attached files.

4. Select the **Remove** check box that corresponds to the file you want to delete.

5. Click **Apply**.

## Configuring discovery

This section describes how BMC Atrium Discovery operates and how to configure and run the product.

The Discovery Engine is designed to locate systems in the network and obtain relevant information from them as quickly as possible, using a variety of different tools and techniques to communicate with devices.

The Reasoning Engine works on the raw data obtained by the Discovery Engine to infer the maximum amount of information about hosts and programs and populate the datastore intelligently. The Reasoning Engine uses patterns that identify running software based on the network ports, processes found, packages installed, protocols used, and so on. It also intelligently searches the discovered data to resolve relationships between items of software.

- Configuring discovery settings
- Masking sensitive data
- Managing the discovery platform scripts
- Adding privileged execution to commands
- Setting up ports for OS fingerprinting

### Configuring discovery settings

Discovery settings can be configured using the Discovery Settings page.

**To set Discovery settings**

1. From the Discovery section of the **Administration** tab, select **Discovery Configuration**.
   Any settings that are not set by default are highlighted by a red change bar.
   The page is grouped into the following sections according to the type of settings:
   - Port Settings
   - Device identification settings
   - Session settings
   - Scanning settings
   - SQL integration settings
   - Other discovery settings

2. Make any required changes to the settings on this page and click **Apply**.

3. To cancel any of your changes, click **Reset To Defaults**.
### Port settings

This section contains settings related to the ports that discovery uses.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP ports to use for initial scan</td>
<td>Enter the TCP ports that will be scanned on a first scan. Use this setting to prevent scanning of any ports that you want to avoid. The default is to use ports: 21, 22, 23, 80, 135, 443, 513, 902 and 3940. See notes on #Order of Operations and TCP and UDP ports to use for initial scan below. Older versions included port 514, which can now be removed from upgraded systems.</td>
</tr>
<tr>
<td>UDP ports to use for initial scan</td>
<td>Enter the UDP ports that will be scanned on a first scan. Use this setting to prevent scanning of any ports that you want to avoid. The default is port 161. See notes on #Order of Operations and TCP and UDP ports to use for initial scan below.</td>
</tr>
<tr>
<td>SSH ports</td>
<td>The default is port 22. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>RLogin ports</td>
<td>The default is port 513. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>Windows ports</td>
<td>The default is port 135. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>Telnet ports</td>
<td>The default is port 23. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>FTP ports</td>
<td>The default is port 21. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>SNMP ports</td>
<td>The default is port 161. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>HTTP ports</td>
<td>The default is port 80. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>HTTPS ports</td>
<td>The default is port 443. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>VMware Authentication Daemon ports</td>
<td>The only supported port is 902. Enter any custom ports to scan in a comma-separated list in the Change To column.</td>
</tr>
<tr>
<td>Mainframe Host Server ports</td>
<td>The default is port 3940. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>WBEM HTTP ports</td>
<td>The default is port 5988. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>WBEM HTTPS ports</td>
<td>The default is port 5989. Enter any custom ports to scan in a comma separated list in the Change To column.</td>
</tr>
<tr>
<td>Valid Port States</td>
<td>When nmap runs port scans, it returns a result of open, closed or filtered. Using the check boxes you can choose which states are valid to investigate further.</td>
</tr>
<tr>
<td></td>
<td>• open/filtered — discover a device that should be accessible but isn’t. It might be open, or filtered.</td>
</tr>
<tr>
<td></td>
<td>• filtered — this port is open but you still cannot connect to it. It must be filtered.</td>
</tr>
<tr>
<td></td>
<td>A port for which a result of open is returned is always considered valid.</td>
</tr>
</tbody>
</table>

Check port 135 before using Windows access methods
**Field Name** | **Details**
--- | ---
Port 135 is usually open on Windows computers. Selecting Yes for this option means that nmap checks whether port 135 is open before a Windows proxy is used to discover an IP device. This is the default. You should select No in firewalled environments where a ping might be filtered by a firewall, but the Windows proxy might be able to connect to the target (for example, it is part of the same Workgroup).

### Device identification settings

This section contains settings related to the methods that discovery uses to identify devices.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use last login method</strong></td>
<td>Discovery uses the discovery method recorded as having been used successfully for an IP address.</td>
</tr>
<tr>
<td><strong>Use SNMP SysDescr to identify OS</strong></td>
<td>Discovery attempts to query the host's SNMP service for the &quot;SysDescr&quot; value to determine the OS.</td>
</tr>
<tr>
<td><strong>Always try &quot;public&quot; SNMP community when using SNMP to identify OS</strong></td>
<td>Discovery attempts to use the public SNMP community to query the host's SNMP service if no credential is available for that host. In this case, only device classification is possible.</td>
</tr>
<tr>
<td><strong>Use Host Server to identify mainframes</strong></td>
<td>Discovery attempts to connect to the host server port to determine whether the discovery target is a mainframe computer.</td>
</tr>
<tr>
<td><strong>Use Telnet banner to identify OS</strong></td>
<td>Discovery makes a telnet connection to a host and uses the telnet &quot;welcome&quot; banner to determine host and OS information.</td>
</tr>
<tr>
<td><strong>Use HTTP(S) HEAD request to identify OS</strong></td>
<td>Discovery attempts to connect to port 80 or 443 of the host and perform an HTTP or HTTPS HEAD request to determine the host and OS.</td>
</tr>
<tr>
<td><strong>Use FTP banner to identify OS</strong></td>
<td>Discovery starts an FTP session with the host and use the FTP &quot;welcome&quot; banner to determine host and OS information.</td>
</tr>
<tr>
<td><strong>Use vSphere API to identify OS</strong></td>
<td>Discovery makes a TCP connection to examine the header and ensure that the VMware authentication daemon is really on port 902 (or the specified port), When confirmed discovery makes a webservices request. This requires open VMware Authentication Daemon and HTTPS port, and a valid vSphere credential.</td>
</tr>
<tr>
<td><strong>Use IP Fingerprinting to identify OS</strong></td>
<td>This option controls whether or not discovery will use IP fingerprinting to determine the OS, if the previous methods have been unsuccessful. The network ports scanned during this phase of discovery can be configured. See Setting Up Ports For OS Fingerprinting. The process of IP fingerprinting can cause instability in some legacy systems, and might trigger intrusion detection systems. This option is enabled by default.</td>
</tr>
<tr>
<td><strong>Use open ports to identify OS</strong></td>
<td>This option controls whether or not open ports are used to identify the OS.</td>
</tr>
</tbody>
</table>

### Session settings

This section contains settings related to the way in which discovery uses sessions to login and run commands.
### Field Name | Details
--- | ---
Session line delay | A delay of 10 ms is introduced between each line sent by Discovery. This avoids problems where remote shells are unable to cope with rapid command sequences. Select one of the following from the list:
  • 1, 2, 5, 10 (the default), 15, 20, 25, 50, 100 milliseconds.
Session login timeout | The length of time for the discovery script to wait for a login prompt. If this is exceeded the attempt is abandoned.
  • Use default timeout.
  • 10, 20, 30, 60, 90, 120, or 180 seconds.
Maximum search window size | The amount of data to examine when detecting the shell command prompt. The default value is 512 bytes. Changing the default value may cause significant degradation of appliance performance. Do not change this value unless directed by Customer Support.
Authorised prompt | Certain systems require an authorization step after logging in. At the command line you are prompted to enter session details. The required response is usually a user name, and some other information.
  In the Authorised Prompt field, enter the text of the prompt.
Authorised response | Where an Authorised Prompt has been entered, you must enter the expected response (that you would enter at the command line) in the Authorised Response field.

### Scanning settings

This section contains settings related to any scanning that discovery undertakes.

| Field Name | Details |
--- | --- |
Ping hosts before scanning | If this option is disabled, then all hosts are discovered, but discovery of empty IP ranges will be slower. The default is to allow discovery to ping the host first.
  If you enable this option, Discovery pings hosts before it starts a scan. Discovery will scan more quickly on empty IP ranges, although hosts might be missed if there are firewalls configured to reject pings. In this situation you should specify IP ranges behind firewalls that you do not want to ping. See the following Exclude ranges from ping option.
  This option only affects scanning of networks other than the one on which the appliance is physically located.
  If you are using ICMP filtering, you should set this option to No.
  See note on #Order of Operations below.
Use TCP ACK ping before scanning | Ping addresses with TCP ACK packets to determine which hosts are actually up. You should use this option when scanning networks that do not permit ping packets. You can specify multiple ports in a comma-separated list.
  This option is only available if the ping hosts before scanning option is set to Yes.
  • See note on #Order of Operations below.
  If Use TCP ACK ping before scanning in your discovery settings is set to Yes, BMC Atrium Discovery may report a device which does not exist on the network rather than dark space (NoResponse) under the following conditions:
  • BMC Atrium Discovery pings an IP address where there is no device, and
  • Some firewall in your environment is configured to respond for that IP address
  To avoid this, it is recommended to either alter such firewall configurations or not to enable TCP ACK ping.
Use TCP SYN ping before scanning | Ping addresses with TCP SYN packets to determine which hosts are actually up. You should use this option when scanning networks that do not permit ping packets. You can specify multiple ports in a comma-separated list.
  This option is only available if the ping hosts before scanning option is set to Yes.
  • TCP SYN pings are likely to trigger IDS and firewall blocks as they are often regarded as ping floods.
  • Also see note on #Order of Operations below.
  If Use TCP SYN ping before scanning in your discovery settings is set to Yes, BMC Atrium Discovery may
Field Name | Details
--- | ---
report a device which does not exist on the network rather than dark space (NoResponse) under the following conditions:
• BMC Atrium Discovery pings an IP address where there is no device, and
• Some firewall in your environment is configured to respond for that IP address
To avoid this, it is recommended to either alter such firewall configurations or not to enable TCP SYN ping.
Exclude ranges from ping | Enter a list of IP addresses or IP ranges that you do not want to ping. For example, you might want to scan IPs which are behind a firewall that blocks ICMP packets. If BMC Atrium Discovery pings an IP address and receives no response, it makes no further attempt to scan that IP address. Excluding a range from pinging enables you to scan IPs behind such firewalls.
This option only affects scanning of networks other than the one on which the appliance is physically located.
Scan retries | Number of retries to be attempted on each host. The system will only retry for machines on which the OS cannot be determined.
The Scan retries and Default OS options work together in sequence to help locate host machines.
Scan timeout | Timeout (in minutes) that applies when BMC Atrium Discovery uses nmap to determine open ports or performs OS fingerprinting. It is not used to limit the time to scan devices. See also the credential timeout for the sessions.
Minimum time before end of window to avoid starting new scheduled discovery operations | A discovery run can take some time to complete. If it is started too close to the end of a Discovery window, it does not complete before the end of the window. To prevent this, you can specify a period in which discovery runs will not be started. The default is 30 minutes, meaning that no discovery runs will be started within 30 minutes of the end of a discovery window. Select the period from the following values in the list:
• 5, 10, 15, 20, 25, 30, 35, 40, and 45 minutes.
Allow scans even if no window defined | Enables you to permit scanning outside permitted discovery windows. The default is no. If you change this option you must restart the tideway service.
Enable running of arbitrary commands | This option controls whether or not arbitrary commands can be run or not. Disabling this option prevents many patterns retrieving information needed to build SIs and BAIs. The default is **Yes**.
Discover Desktop Hosts | Use this option to permit or prevent discovery of desktop hosts. The default is No, that is, do not discover desktop hosts. When the option is set to No, if a Windows or Mac OS host is determined to be desktop then the host is skipped. An SMB query is used to determine whether the endpoint is a Windows desktop. See [Dark space scanning](Dark space scanning) for more information. When a host is skipped, the device_type attribute on the Device Info Node is set to Desktop, no inferred host is created, and the corresponding DiscoveryAccess result is shown as "Skipped (Desktop host discovery has been disabled)".
Discover neighbor information when scanning network devices | Cause discovery to retrieve MAC and port information from neighboring scanned network devices. The default is Yes. Only select No if you do not want to collect any edge connectivity information.

**SQL integration settings**

This section contains settings related to SQL integrations.
### Field Name | Details
--- | ---
Timeout to establish a connection | The timeout for establishing a connection to the database. Select the timeout period in seconds from the following values in the list:
  - 5, 10, 30 (the default), 60, 90, 120, and 180.

Maximum connections held open | Specifies the number of connections to databases that can be held open after they would otherwise be closed. Higher values can reduce connection delays but will consume extra resources. The default is unlimited. If you change this option you must restart the tideway service. Select the number of connections from the following values in the list:
  - 0, 10, 20, 30, 40, 50, and Unlimited.

Maximum time to hold an unused connection open | Specifies the maximum time to hold an unused database connection open. Higher values can reduce connection delays but will consume extra resources. The default is 2 minutes. If you change this option you must restart the tideway service. Select the timeout period in minutes from the following values in the list:
  - 2, 4, 6, 8, and 10.

---

### Mainframe discovery

This section contains settings related to mainframe discovery.

### Field Name | Details
--- | ---
Timeout to establish a connection | The timeout for establishing a connection to the database. Select the timeout period in seconds from the following values in the list:
  - 30, 60, 90, 120 (the default), 180, and 300.

---

### Other discovery settings

This section contains other discovery settings.

### Field Name | Details
--- | ---
Recording mode | Record and playback modes are intended for diagnostic support and testing. Select the discovery mode from the list, it can be one of the following:
  - Off — normal type of discovery in which the appliance scans IP ranges on the network, runs scripts on targets, and uses Reasoning to process the results. In this mode, pool data is not created. This is the default.
  - Record — record mode is the same as Normal mode but in addition, the raw discovered data is stored on the appliance so that it can be used in Playback mode. In this mode pool data is created in addition to record data. Creating record and pool data imposes considerable overhead on the system and is rarely needed.
  - Playback — in Playback mode, data that has been recorded in Record mode is used to replay discovery. In this mode, Discovery does not scan any targets on the network.

Maximum concurrent discovery requests per engine | Specifies the maximum number of concurrent discovery requests permitted per processing engine. The maximum value and available range of settings is calculated for optimum performance depending on the appliance. Values shown in the list depend upon the number of processing engines. The base values in the list are:
  - 30 (default), 60, 90, 120, or 150.
Values shown will be the base values multiplied by the number of processing engines.
If you change this option you must restart the tideway service.
You should leave this setting at its default unless you are experiencing many discovery commands timing out. As a general rule, for more discovery requests permitted concurrently, you increase network traffic, and the absolute time for discovering a single host increases. However, with the interleaving of discovery processing, total throughput might be improved.
Trade discovery performance with interactivity

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
|                             | Scanning IP addresses, consolidation and manual pattern execution can place a heavy load on the system. During such times the UI can become unresponsive. You can choose to delay these tasks to provide additional resources for the UI. This provides better interactivity with the system, but at a cost in raw discovery performance. You can choose from the following:
|                             | • Prefer discovery — the default and the behavior of previous versions of BMC Atrium Discovery.
|                             | • Balance both
|                             | • Prefer interactivity
|                             | If you change this option you must restart the tideway service.                                                                                                                                         |

Minimum Windows Proxy version

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
|                             | The minimum version of the Windows proxy that the appliance will use for Windows Discovery. You can enter a new minimum Windows proxy version in this field. Ensure that you do not include any whitespace in the version number. The version number of a Windows proxy corresponds to the version number of BMC Atrium Discovery that the Windows proxy was released with (for example, 9.0).
|                             | **Note:** You must stop scanning to change the minimum Windows proxy version or release.                                                                                                |

Enable Automatic Grouping

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Automatic Grouping is the automatic grouping of hosts into logical groups called Automatic Groups. This is primarily intended to help in baselining. By default it is enabled. Select this option to enable Automatic Grouping. Disabling Automatic Grouping might improve scanning performance.</td>
</tr>
</tbody>
</table>

Scanner File polling interval

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
|                             | Scanner files are used to simulate discovery of inaccessible hosts. Discovery polls for new scanner files periodically. Select the polling interval from the following values in the list:
|                             | • Every minute, Every hour, and Every day.
|                             | If you change this option you must restart the tideway service.                                                                                                                                         |
|                             | **Note:** When set to Every day, the polling time is at midnight UTC time. Daylight saving time is not considered.                                                                                      |

Scanning and data processing levels

The Discovery Engine and the Reasoning Engine, collectively the discovery process, cooperate to:

- scan selected ranges
- process the data
- create a detailed data model

The data model defines all discovered objects and the relationships between them and is defined in the system Taxonomy.

The processing that the discovery process uses to create this complex, detailed, and interrelated data model is considerable. You can control the level of processing used, and consequently the accuracy, complexity, and detail in the data model. This gives performance benefits at a cost of model accuracy. You might find that reducing the level of processing used speeds rollout of BMC Atrium Discovery throughout your organization.

The following levels are available:

- **Sweep Scan** — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It attempts to log in to a device to determine the device type.
- **Full Discovery** — Retrieves all the default information for hosts, and complete full inference.
Order of operations

Where selected, these groups of operations are carried out in the following order:

- Ping hosts before scanning
- Use TCP ACK "ping" before scanning
- Use TCP SYN "ping" before scanning
- TCP ports to use for initial scan
- UDP ports to use for initial scan
- Use IP Fingerprinting to Identify OS

TCP and UDP ports to use for initial scan

The initial port scan is an important part of discovery. If you remove a port from the initial port scan, that port is effectively removed from discovery. For example, if you remove port 22, you will effectively disable ssh access.

The Use Last Login settings override any settings made in ports for use for initial scan. For example, if you disable port 23 using this feature, but a host has previously been discovered using telnet, this host is still discovered using telnet, because it is listed as the last login for that host.

Masking sensitive data

The View Object page of a Discovered Process shows the command used to start the process. In some cases, a user name and password, or some other sensitive data is shown in clear text. You can also view the contents of a discovered file, and in some cases these too can contain passwords or other sensitive data. You can prevent this using Sensitive data filters.

Sensitive data filters for processes only mask information from the discovered process or file; not from, for example, package names.

A sensitive data filter is a regular expression to define data that you do not want displayed. When matched, the sensitive portion of the data is hashed using MD5. The hashed data can be compared with earlier versions to determine whether it has changed, while the actual data remains hidden from users.

ён Sensitive data filters use MD5 hash

Common passwords and dictionary words can be extracted from MD5 hashes using commonly available tools. If you rely on sensitive data filters to entirely mask passwords, you should ensure that any that may appear in discovered data are good strong passwords.
Managing Sensitive Data Filters

1. From the Discovery section of the Administration tab, select Sensitive Data Filters. The Sensitive Data Filters window is displayed with the **Processes** tab visible.

2. To view or edit filters for files, click the **Files** tab.
3. To edit an existing filter, click **Edit**.
4. To delete an existing filter, click **Delete**.
5. To add a new filter, click **Add ...** A new field is added into which you can enter a regular expression.
6. To create the filter, click **Apply**.

To reorder sensitive data filters, click the up or down arrow in the ordering column for the filter you want to move. You can also move a filter to the top or bottom of the list using the top or bottom arrow buttons.

Creating a Sensitive Data Filter

The regular expression will usually match more than just the sensitive data, including for instance an identifying argument name like "--password". The portion of data to be hashed must be enclosed in brackets to form a regular expression group. Portions of the regular expression not enclosed in the brackets will be unmodified.

1. The following command has the "--password" in clear text. The regular expression needs to use "--password" to locate the data, and define how much to mask around it.

   ```
   ./pfg_serv -h -Hj -g lob -l full --user gussie --password finknottle --dominion emea
   ```

2. This regular expression adds "\S+" to identify a sequence of one or more non whitespace characters, making a regular expression of "--password \S+". Brackets are then added to define the portion that requires masking, making "--password (\S+)".
3. After rediscovery, the new process node will have the password portion replaced with an md5 hash.

   ```
   ./pfg_serv -h -Hj -g lob -l full --user gussie
   --password 4343ab718997a9570ab20c0c1b5e18ad --dominion emea
   ```
4. For more resilience against extra white space the single space in the regular expression should be replaced with `\s+`, which matches any whitespace character making "--password\s+(\S+)" which is the form that most sensitive data filters would take.

Notes on Sensitive Data Filters

1. When writing regular expressions for sensitive data filters, you should ensure that it does not match too much of the command. If the filter masks some of the command that a pattern uses to identify a piece of running software, that pattern will then be unable to identify the software. See Writing efficient regular expressions for more information.
2. The filters are not applied to the inferred data model until you perform a discovery run. Sensitive data discovered before applying a filter will remain in the history and DDD until it is aged.
3. If applied to files, the files must remain valid. For example, if applied to an XML file, the XML must remain valid otherwise Xpath processing will not work.

Managing the discovery platform scripts

The discovery system runs various scripts through the login session to obtain information from the host systems. From the BMC Atrium Discovery user interface, you can:

- View the scripts
- Amend scripts (edit script parameters, or enable or disable scripts)

However, the BMC Atrium Discovery user interface does not enable you to add new scripts.

Where this page refers to "default scripts", it refers to the discovery scripts that were shipped with the release. For example, in a version 9.0.02 appliance, the default scripts are those that shipped with 9.0.02. If you upgrade the appliance to version 10.0, the default scripts are those that shipped with version 10.0.

If you have just upgraded your installation and have not updated the discovery scripts, you can use the Discovery Platforms page to view the script differences and choose to upgrade to the current version by clicking Reset to Default. This enables you, post-upgrade, to view the script differences and choose to upgrade to the current version by clicking **Reset to Default**.

Viewing the existing scripts

To view existing scripts, perform the following:

1. From the Discovery section of the Administration tab, select **Platforms**.
   The **Discovery Platforms** page lists the categories for which discovery commands are provided.
   The available categories are:
   - Windows Discovery
   - UNIX, Linux and Related Discovery
Mainframe Discovery

SNMP Supported Platforms

Each category displays headers which link to pages containing details of discovery methods used. Some additional information associated with a header might be displayed as a tool tip. The following table summarizes the available discovery methods in these pages:

<table>
<thead>
<tr>
<th>Category</th>
<th>Discovery method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows discovery</td>
<td>The discovery methods used are WMI, RemQuery, and SNMP.</td>
</tr>
<tr>
<td></td>
<td>Windows discovery uses WMI and RemQuery as the primary method of discovery. Even though two different techniques are used as the primary method of discovery, it is considered to be a single session. If WMI and RemQuery access fails to establish the initial session, Windows discovery attempts to use SNMP. Once an initial session is established, Discovery does not attempt another session (for example, if WMI and RemQuery establishes the initial session but discovery is incomplete, it does not attempt another session using SNMP).</td>
</tr>
<tr>
<td>UNIX, Linux &amp; Related Discovery</td>
<td>The discovery methods used are Shell Scripts and SNMP. For VMware ESX and ESXi platforms, vSphere API calls are used.</td>
</tr>
<tr>
<td></td>
<td>UNIX, Linux and related discovery uses an interactive login (ssh, telnet, or rlogin) and running shell scripts as the the primary method of discovery (in case of VMware ESX and ESXi platforms, vSphere API calls are used as the the primary method of discovery). If the primary method of discovery fails to establish the initial session, Discovery attempts SNMP. Once an initial session is established, Discovery does not attempt another session (for example, if interactive login and running shell scripts establish the initial session and even if some scripts fail or commands are not successfully executed, Discovery does not attempt another session with SNMP).</td>
</tr>
<tr>
<td>Mainframe</td>
<td>Discovery uses the MainView Host Server to obtain mainframe information.</td>
</tr>
<tr>
<td>SNMP supported platforms</td>
<td>SNMP discovery is supported for all the devices with an accessible SNMP agent. Discovery supports SNMP v1, v2c and v3.</td>
</tr>
</tbody>
</table>

2. Where the discovery scripts have been changed from the default, a change bar is shown to the left of the platform name.

3. To manage the changed scripts in the UNIX, Linux and Related Discovery section, you can use the following controls:
   - **Reset All** — click this to reset all scripts to the default.
   - **View Script Differences** — this link opens a page showing a list of changed discovery methods, sorted by platform.

Click the discovery method header to reveal the detailed view of the differences between the default method and the current method:
4. Click the OS link whose commands you want to edit. You cannot edit the SNMP methods, you can only view them.

**Amending the existing scripts**

You can only edit and enable or disable selected shell scripts from the Discovery Platforms page. You can only view and not amend the following methods:

- WMI
- RemQuery
- SNMP
- vSphere
- Mainframe

To amend existing scripts, perform the following:

1. On the Discovery Platforms page, click the OS link corresponding to the commands you want to edit.
2. Click on the Shell Scripts tab.

The commands for the OS are displayed. The following table describes the fields in this page:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download host script</td>
<td>To download the entire script, click this link.</td>
</tr>
<tr>
<td>Path</td>
<td>To edit the command path which is set when a shell is opened on the target, click the corresponding Edit link. Just the command path displayed is set, it is not added to any existing path in the user environment.</td>
</tr>
<tr>
<td>Show All Scripts</td>
<td>To display a view only version of all the scripts, click this link.</td>
</tr>
<tr>
<td>Method</td>
<td>Displays the discovery method used. * indicates methods that will successfully create a host.</td>
</tr>
<tr>
<td>Script</td>
<td>To display a view only version of a script, click this link for the corresponding script. Some scripts might be marked as Privileges. This means that a normal login does not have sufficient access rights to get all of the information required. In this case, the root, or another user with higher access rights is used. Displays if the script is selected to be enabled or not.</td>
</tr>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
</tbody>
</table>
| Enabled or Disabled | In the Actions column, the following options are available:  
  - **Edit** — To edit the script, click this link. You can edit the discovery script, notes, and enable or disable the script.  
  - **Disable or Enable** — To change the enable and disable status of a script, click this link.  
  - **Download** — To download a copy of the script, click this link. You can then edit it in a text editor and save it locally.  
  - **Upload** — To upload a script to replace the existing one, click this link.  
    Where you edit or replace an existing script, a red change bar shows that the script has been changed and an additional **Reset To Default** link is displayed. Click this link to reset the script to the default. |

**Maximum Line Length**

The maximum line length of an interactive script is OS and shell dependent, and exceeding such a limit will cause the commands to fail. Ensuring that all lines in discovery scripts contain fewer than 128 characters will prevent this occurring.

3. Click **Apply** to save the changes.

**Additional discovery**

In addition to the standard discovery commands described above, additional discovery methods can be called by patterns. See [manual pattern execution](#) and [DiscoveryAccess page](#).

Methods such as the following are used:

- **getFileInfo**: Used where a pattern uses the `discovery.fileGet` to return a file. The result is stored in a `DiscoveredFile` node.
- **getFileInfo**: Used where a pattern uses the `discovery.fileInfo` to return the metadata on a file. The result is stored in a `DiscoveredFile` node with no `content` attribute.
- **runCommand**: Used where a pattern uses the `discovery.runCommand` function to run a command on a host during its execution. The result is stored in a `DiscoveredCommandResult` node.
- **getDirectoryListing**: Used where a pattern uses the `discovery.listDirectory` function to return a list of the directory contents. Returns a list of `DiscoveredDirectoryEntry` nodes.
Returning command exit status data

The commands that Discovery runs on UNIX hosts are all standard UNIX commands, and can be viewed through the user interface. When Discovery runs these commands they return an exit status code. Discovery provides the ability to record exit status codes and output to stderr where commands fail. Success is not recorded.

You should bear in mind standard scripting techniques when editing these commands. For example, some Discovery commands use pipelines, and might need to be modified so as not to mask the desired exit status code.

Exit status codes are OS, OS version, and command dependent. Consult your OS documentation for information on exit status codes and their meanings.

Recording the exit status

To record the exit status you need to prefix the discovery commands with `tw_capture`. The first parameter is a name used to identify the captured information. The second parameter is the command to run, and any further parameters are passed to the command. For example:

```
tw_capture fileinfo_foo ls -ld /usr/bin/foo
```

In order to be able to capture output using `tw_capture`, `/tmp` on the target host needs to be writable by the discovering account.

You can record an exit status from commands used in a pipeline, for example:

```
tw_capture foo dpkg -l '*' | egrep '^ii '
```

It can also be used in backticks, for example when running a command and assigning the result to a shell variable:

```
dns_domain=`tw_capture hostname -d | sed -e 's/(none)//'`
```

The following screen shows two example `tw_capture` commands, one which will succeed, and one which will fail. These have been added to the start of the `getDeviceInfo` method:
**Viewing the exit status**

After you have added the `tw_capture` commands and discovered a host causing the method to be run, you can see any script failures by viewing the appropriate method link on the Discovery Access page.

To view the exit status:

1. From the Discovery Recent Runs tab, click the discovery run which you used to scan the host.
2. Click **DiscoveryAccess**.
   - If it is a link to a single DiscoveryAccess, then that DiscoveryAccess page is shown. If there are multiple DiscoveryAccesses, then a list page is displayed.
3. Click the method link for the method which you edited to add the return status.

**Adding privileged execution to commands**

BMC Atrium Discovery is shipped without any commands that use privileged execution. The following example illustrates how to add privileged execution to `lsof` commands. You must do the same for any command that you want to execute as a privileged user. The individual discovery scripts for each platform, `getDeviceInfo`, `getFileSystems`, and so on, use a privilege mechanism which is configured in the `initialise` script. You do not need to edit the individual scripts to escalate privileges, only the `initialise` script.

To configure execution of a command as a privileged user:

1. In the Discovery section of the Administration page, click **Platforms**.
2. Click the OS link corresponding to the commands on which you want to add the privileged execution.
3. Click **Edit** in the Action column of the `initialise` method row.
   - The edit window is shown containing the script.
4. Click in the edit window to enlarge it, as illustrated in the following screen.
5. Use the find function in your browser to search for the PRIV section (search for PRIV_XXX to find the beginning of the PRIV section).

6. In the PRIV function (in this example PRIV_RUNCMD), add the command required (such as sudo, pbrun, or dzdo) to run the commands as a privileged user.

   For example:

   ```bash
   ... 
   PRIV_RUNCMD() {
   sudo "$@
   }
   ...
   ```

   Alternatively, if you need to specify the path:

   ```bash
   ... 
   PRIV_RUNCMD() {
   /usr/bin/sudo "$@
   }
   ...
   ```

   You can also limit the privilege escalation to a particular command, in this case pmap:

   ```bash
   ... 
   PRIV_RUNCMD() {
   if [ $1 = "pmap" or $1 = "usr/bin/pmap" ]; then
   sudo "$@
   else
   "$@
   fi
   } 
   ... 
   ```
7. Click **Apply** to apply the changes. The screen is refreshed and the initialise method is highlighted to show that it has changed from the default.

8. Click **Show Differences** to show the differences between the default script and the current script.

The $@ represents the command that BMC Atrium Discovery issues. Adding `sudo` (or similar privileged command) tells it how to escalate the privilege for that command. Now when a script needs to call `pmap`, it calls the PRIV_RUNCMD() command with the full command it needs to run, which then runs `pmap` with the correct privilege.

⚠️ If the path is specified, it will affect all discovery commands that use that function. The privileged command might not always be at the same place on all discovery targets.

If the path is not specified, the privileged command will be found with the path of the user profile and the BMC Atrium Discovery path environment variable. You can check the path environment variable as it is displayed at the top of the Platforms page.

You must add a privileged execution method to whichever commands you require in order to gain the fullest possible discovery. The available commands, their impact on discovery and the platforms they are available on described on the Privileged commands page.

**Password prompt in privileged command execution**

Where the sudo (or similar privileged command) configuration on a target host requires the user password to be entered at the command line, discovery will resend the credential already used to log on to the target.

In such situation however, if the default sudo "Password:" prompt has been customized on target systems (for example, by setting the SUDO_PROMPT environment variable, or specifying a passprompt entry in the target's sudoers file), then the initialize script for the corresponding platform would need to be edited to specify:
Privileged commands in Solaris

Solaris versions 9 and later no longer use sudo as the preferred method of privilege escalation, rather, they use a more sophisticated Role Based Access Control (RBAC) privilege mechanism. One of the ways of granting a user escalated privileges is to assign them a role, which can be either system, or user defined. The preferred way to provide escalated privileges for BMC Atrium Discovery is to grant the proc_owner role to the discovery user. This enables the discovery user to obtain information on processes that belong to other users.

An alternative method is to use elevated profiles using the pfexec command. This prompts for a password, but will be handled by the discovery scripts in the same way as sudo.

Privileged commands

This page describes the available privileged commands, their impact on discovery, and the platforms on which they are available. By default, each command is left unprivileged (for example, `PRIV_LSOF()`). The user or administrator must modify the script to insert the relevant command to allow discovery to run the privileged commands. Examples are provided in adding privileged execution to commands.

AIX

PRIV_CAT: The cat command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.
PRIV_LSOF: The lsof command requires superuser privileges to display information on processes other than those running as the current user.
PRIV_RUNCMD: This function supports running privileged commands from patterns.
PRIV_LSLPP: The lslpp command requires superuser privileges to list all installed packages.
PRIV_TEST: This function supports privilege testing of attributes of files.
PRIV_LS: This function supports privilege listing of files and directories.

FreeBSD

PRIV_CAT: The cat command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.
PRIV_LSOF: The lsof command requires superuser privileges to display information on processes other than those running as the current user.
PRIV_RUNCMD: This function supports running privileged commands from patterns.
**PRIV_DMIDECODE:** The `dmidecode` command requires superuser privileges to read data from the system BIOS.

**PRIV_TEST:** This function supports privilege testing of attributes of files.

**PRIV_LS:** This function supports privilege listing of files and directories.

**HPUX**

**PRIV_CAT:** The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF:** The `ls/of` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD:** This function supports running privileged commands from patterns.

**PRIV_LANADMIN:** The `lanadmin` command requires superuser privileges to display any interface speed and negotiation settings.

**PRIV_SWLIST:** The `swlist` command requires superuser privileges to list all installed packages.

**PRIV_TEST:** This function supports privilege testing of attributes of files.

**PRIV_LS:** This function supports privilege listing of files and directories.

**FCMSUTIL:** This command requires superuser privileges to list attributes of HBA devices.

**IRIX**

**PRIV_CAT:** The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF:** The `ls/of` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD:** This function supports running privileged commands from patterns.

**PRIV_TEST:** This function supports privilege testing of attributes of files.

**PRIV_LS:** This function supports privilege listing of files and directories.

**Linux**

**PRIV_CAT:** The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF:** The `ls/of` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD:** This function supports running privileged commands from patterns.

**PRIV_TEST:** This function supports privilege testing of attributes of files.

**PRIV_LS:** This function supports privilege listing of files and directories.

**PRIV_DMIDECODE:** The `dmidecode` command requires superuser privileges to read data from the system BIOS.

**PRIV_HWINFO:** The `hwinfo` command requires superuser privileges to read data from the system BIOS

**PRIV_MII TOOL:** The `mii-tool` command requires superuser privileges to display any interface speed and negotiation settings.

**PRIV_ETHTOOL:** The `ethtool` command requires superuser privileges to display any interface
speed and negotiation settings.

**PRIV_NETSTAT**: The `netstat` command requires superuser privileges to display process identifiers (PIDs) for ports opened by processes not running as the current user.

**PRIV_LPUTIL**: The `lputil` command requires superuser privileges to display any HBA information.

**PRIV_HBACMD**: The `hbacmd` command requires superuser privileges to display any HBA information.

**PRIV_XE**: The `xe` command requires superuser privileges to report CPU information on Xen platforms.

**PRIV_ESXCFG**: The `esxcfg-info` command requires superuser privileges to report hardware information on a VMWare ESX controller.

**Mac OS X**

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**NetBSD**

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**PRIV_DMIDECODE**: The `dmidecode` command requires superuser privileges to read data from the system BIOS.

**OpenBSD**

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**PRIV_DMIDECODE**: The `dmidecode` command requires superuser privileges to read data from the system BIOS.
OpenVMS
Not applicable to this platform. The Normal privilege category is sufficient to run the commands in the discovery script.

POWER HMC
Not applicable to this platform.

Solaris
Solaris versions 9 and later no longer use sudo as the preferred method of privilege escalation, rather, they use a more sophisticated Role Based Access Control (RBAC) privilege mechanism. One of the ways of granting a user escalated privileges is to assign them a role, which can be either system, or user defined. The preferred way to provide escalated privileges for BMC Atrium Discovery is to grant the proc_owner role to the discovery user. This enables the discovery user to obtain information on processes that belong to other users.

An alternative method is to use elevated profiles using the pfexec command. This prompts for a password, but will be handled by the discovery scripts in the same way as sudo.PRIV_CAT: The cat command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.
PRIV_LSOF: The lsof command requires superuser privileges to display information on processes other than those running as the current user.
PRIV_RUNCMD: This function supports running privileged commands from patterns.
PRIV_TEST: This function supports privilege testing of attributes of files.
PRIV_LS: This function supports privilege listing of files and directories.
PRIV_EMLXADM: The emlxadm command requires superuser privileges to display any HBA information.
PRIV_FCINFO: The fconfig command requires superuser privileges to display any HBA information.
PRIV_DMI: The dmidecode command requires superuser privileges to read data from the system BIOS on Solaris X86 platforms only.
PRIV_IFCONF: The ifconfig command requires superuser privileges to display the MAC address of each # interface.
PRIV_NDD: The ndd command requires superuser privileges to display any interface speed and negotiation settings.
PRIV_PS: The /usr/ucb/ps command requires superuser privileges to display full command line information (without this, command lines will be limited to 80 characters). This affects Solaris 10 and later and Solaris 8 & 9 with certain patches.
PRIV_LPUTIL: The lputil command requires superuser privileges to display any HBA information.
PRIV_HBACMD: The hbacmd command requires superuser privileges to display any HBA information.
PRIV_PFILES: The pfiles command requires superuser privileges to display open port...
information for processes not running as the current user.

**PRIV_DLADM**: The `dladm` command requires superuser privileges to display any interface speed and negotiation settings.

**Tru64**

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_SETLD**: The `setld` command requires superuser privileges to display information on installed packages.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**UnixWare**

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**VMware ESX**

This refers to ssh discovery rather than discovery via the vSphere API.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

**PRIV_DMIDECODE**: The `dmidecode` command requires superuser privileges to read data from the system BIOS.

**PRIV_HWINFO**: The `hwinfo` command requires superuser privileges to read data from the system BIOS.

**PRIV_MIITool**: The `mii-tool` command requires superuser privileges to display any interface speed and negotiation settings.

**PRIV_ETHTOOL**: The `ethtool` command requires superuser privileges to display any interface speed and negotiation settings.

**PRIV_NETSTAT**: The `netstat` command requires superuser privileges to display process...
identifiers (PIDs) for ports opened by processes not running as the current user.

**PRIV_LPUTIL**: The `lputil` command requires superuser privileges to display any HBA information.

**PRIV_HBACMD**: The `hbacmd` command requires superuser privileges to display any HBA information.

**PRIV_XE**: The `xe` command requires superuser privileges.

**PRIV_ESXCFG**: The `esxcfg-info` command requires superuser privileges.

**VMware ESXi**

This refers to ssh discovery rather than discovery via the vSphere API.

**PRIV_LSOF**: The `lsof` command requires superuser privileges to display information on processes other than those running as the current user.

**PRIV_RUNCMD**: This function supports running privileged commands from patterns.

**PRIV_CAT**: The `cat` command requires superuser privileges to display the contents of files not readable by the current user. For example, configuration files owned by the root user.

**PRIV_TEST**: This function supports privilege testing of attributes of files.

**PRIV_LS**: This function supports privilege listing of files and directories.

### Setting up ports for OS fingerprinting

BMC Atrium Discovery maintains a list of the port numbers that it interrogates. These ports are only scanned if the "Use IP Fingerprinting to Identify OS" option is enabled. See [Configuring discovery settings](#) for more information on this option.

If you want to set up any additional ports, you need to enter this here (for example if you use a non-standard port number).

### To set up port settings

1. To locate the **Port Scanning** page, navigate to the **Discovery Configuration** page in the Discovery section of the Administration tab. Next to the **Use IP Fingerprinting to Identify OS** option you will find the Configure Port Scanning link.

2. The Port Scanning page is displayed.

   Any currently configured ports are listed.

3. Edit the port settings:
   - To create a new port value, click **Add**.
     
     An additional entry row is displayed at the end of the table.
   - To amend an existing value, click **Edit** for that entry.
   - To remove an entry, click **Delete**.

4. Set up the values as follows:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>Port number to be used.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of application.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of the application.</td>
</tr>
<tr>
<td>TCP Enabled</td>
<td>Select this check box if TCP uses this port assignment.</td>
</tr>
<tr>
<td>UDP Enabled</td>
<td>Select this check box if UDP uses this port assignment.</td>
</tr>
<tr>
<td>Reason</td>
<td>Complete this if a non-standard port has been used.</td>
</tr>
</tbody>
</table>

5. Click **OK** to save the new or changed details.
6. Click **Apply** to remove the lock and allow other users to edit the settings. If you click **Cancel** at this point, any changes you have made are discarded.

## Running discovery

The following topics provide information and instructions for running and monitoring BMC Atrium Discovery:

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing discovery status</td>
<td>This topic describes the available information about a discovery process.</td>
</tr>
<tr>
<td>Starting and stopping discovery</td>
<td>This topic lists the discovery statuses and explains how to manage a discovery process.</td>
</tr>
<tr>
<td>Scanning IP addresses or ranges</td>
<td>This topic describes the procedure for setting up an immediate or scheduled discovery and for enabling/disabling a discovery run.</td>
</tr>
<tr>
<td>Excluding IP addresses and ranges from scanning</td>
<td>This topic shows how to avoid scanning specific IP addresses (either permanently or on schedule).</td>
</tr>
<tr>
<td>Stalled discovery runs</td>
<td>This topic explains the behavior of the discovery runs in context of the scan windows.</td>
</tr>
<tr>
<td>Edge connectivity</td>
<td>This topic describes the process and consequences of enabling/disabling discovery of the neighbor information from network devices.</td>
</tr>
</tbody>
</table>

### Viewing discovery status

The status of the discovery process displays on the **Home** tab in the Discovery Status summary. This page also displays the current status of the reasoning process.

- To view discovery status
- Consolidation
- Currently processing runs
- Recent runs
- Scheduled runs
- Exclude ranges

**To view discovery status**

The Discovery Status page is the central point for running discovery. It contains the following tabs:

- Currently Processing Runs shows the first 10 runs that are currently in progress.
Recent Runs shows the 10 most recently completed runs.
Scheduled Runs lists the scheduled runs.
Exclude Ranges shows the first 10 excluded ranges and enables you to add new exclude ranges.

To display the Discovery Status page, click Discovery is currently next to the green RUNNING or red STOPPED button. The Discovery Status page also provides a START ALL SCANS button or a STOP ALL SCANS button, depending on whether BMC Atrium Discovery is running or not.

**Note**

By default, the discovery process is off. To start it, you must click START ALL SCANS.

The default view of the Discovery Status page, with a snapshot scan in progress and a scheduled scan on hold, is shown in the following screen:

There are record and playback modes which are intended for diagnostic support and testing. If one of these is in operation, a label is displayed the right hand side of the page. See discovery settings for more information.

**Consolidation**

An appliance can be configured to be either a Consolidation appliance or a Scanning appliance. Consolidation refers to the centralization of discovery data from scheduled or snapshot scans on multiple Scanning appliances to one or more Consolidation appliances. For a detailed description about consolidation and how to configure it, see Consolidation.
Currently processing runs

The **Currently Processing Runs** tab shows the first 10 discovery runs that are in progress. The tab contains the following fields:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>The descriptive label that is applied to the run. This is the label applied by the user who created the run.</td>
</tr>
<tr>
<td>Started</td>
<td>The scan start time. This is the time that the discovery run was requested rather than the actual start time. In practice these are almost identical; the only time that a large discrepancy is seen is when the discovery Run is requested at the end or outside of a scan window.</td>
</tr>
<tr>
<td>Range</td>
<td>The IP range that is being scanned in this discovery run. For a large list of IP addresses this shows the first two IP addresses and the number of other IP addresses in the queue.</td>
</tr>
<tr>
<td># of IPs</td>
<td>The total number of IP addresses being scanned in this discovery run.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of scan. This is either: • Snapshot • Scheduled</td>
</tr>
<tr>
<td>Level</td>
<td>The level for the discovery run. This is one of: • Sweep Scan — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type. • Full Discovery — Retrieves all the default info for hosts, and complete full inference.</td>
</tr>
<tr>
<td>Percent Complete</td>
<td>A progress indicator showing how much of the scan has been competed and how many are in progress.</td>
</tr>
<tr>
<td>User</td>
<td>The user name of the user who created the scan.</td>
</tr>
</tbody>
</table>

If a discovery run is in progress but has been stopped because the scan window in which it can take place has ended, it is *On Hold*. On hold scans are shown in the **Currently Processing Runs** tab and marked as such.

You can select a discovery run and cancel it using **Cancel**. The discovery run will stop though individual commands in progress will run to completion.

Adding new discovery runs is described in [Scanning IP addresses or ranges](#).

**Recent runs**

The **Recent Runs** tab shows the 10 most recently completed discovery runs, and contains the following fields:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>The descriptive label that is applied to the run. This is the label applied by the user who created the run.</td>
</tr>
<tr>
<td>Range</td>
<td>The IP range that was scanned in this discovery run.</td>
</tr>
<tr>
<td># of IPs</td>
<td>The total number of IP addresses scanned in this discovery run.</td>
</tr>
<tr>
<td>Timing</td>
<td></td>
</tr>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>The scan start time, finish time, and elapsed time. This is the time that the discovery run was requested rather than the actual start time. In practice these are almost identical; the only time that a large discrepancy is seen is when the discovery run is requested at the end or outside or a scan window.</td>
<td></td>
</tr>
<tr>
<td>Complete</td>
<td>States whether the scan was completed or cancelled by the user.</td>
</tr>
</tbody>
</table>
| Type | The type of scan. This is either:  
  • Snapshot  
  • Scheduled |
| Level | The level for the discovery run. This is either:  
  • **Sweep Scan** — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type.  
  • **Full Discovery** — Retrieves all the default info for hosts, and complete full inference. |
| User | The name of the user who created the scan. |
| Actions | A **Rescan Now** button is provided in this column which enables you to rescan the discovery run. If you rescan a scheduled scan, it is scanned immediately as a snapshot scan with the same label. |

**Scheduled runs**

The **Scheduled Runs** tab shows up to 10 scheduled discovery runs, and contains the following fields:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection column</td>
<td>For each row in the <strong>Scheduled Runs</strong> tab, a checkbox is provided enabling you to select individual scheduled runs. Selecting the checkbox in the heading row selects all runs in the tab.</td>
</tr>
<tr>
<td>Label</td>
<td>The descriptive label that is applied to the run. This is the label applied by the user who created the run.</td>
</tr>
<tr>
<td>IP Range</td>
<td>The IP range to be scanned in this discovery run.</td>
</tr>
</tbody>
</table>
| Level | The level for the discovery run. This is either:  
  • **Sweep Scan** — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type.  
  • **Full Discovery** — Retrieves all the default info for hosts, and complete full inference. |
| Date Rules | Describes the schedule for this discovery run. The days on which the discovery run takes place, the time it starts, and the period in which the scan is permitted to run. |
| State | The scan start time. |
| Enabled | Whether the scheduled run is currently enabled or disabled. **Yes** for enabled, **No** for disabled. |
| User | The name of the user who created the run. |
| Created | The date that the scheduled run was created. |

You can enable, disable, or delete selected scheduled runs using the **Enable**, **Disable**, or **Delete** buttons at the foot of the table. See **scheduled runs** for more information on enabling and disabling scheduled runs. Disabling a scheduled run that is in progress cancels that run.

Adding scheduled discovery runs is described in **Scheduling discovery runs**.
Exclude ranges

The Exclude Ranges tab shows the ranges which have been specified as not to be scanned, and contains the following fields:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection column</td>
<td>For each row in the Exclude Ranges tab, a checkbox is provided enabling you to select individual scheduled runs. Selecting the checkbox in the heading row selects all exclude ranges in the tab.</td>
</tr>
<tr>
<td>Label</td>
<td>The descriptive label that is applied to the exclude range.</td>
</tr>
<tr>
<td>Range</td>
<td>The IP range that is not to be scanned in any discovery run.</td>
</tr>
<tr>
<td>Description</td>
<td>A description for the exclude range.</td>
</tr>
<tr>
<td>User</td>
<td>The user who created the exclude range.</td>
</tr>
</tbody>
</table>

You can delete selected exclude ranges using the Delete button at the foot of the table.

Adding new exclude ranges is described in Exclude Ranges.

Starting and stopping discovery

The status of the discovery process is shown at the top of the Discovery Status page. The discovery process can be in the following statuses:

- Running
- Stopping
- Stopped

When you start the discovery process, any runs that are on hold or waiting start transition into In progress. If a run is on hold but outside its scheduled window, it does not start.

When you stop the discovery process, any regular or snapshot runs currently in progress stop. The status changes to Stopping while the discovery process is stopping.

To start the discovery process

Click START ALL SCANS.

The screen is automatically refreshed to show the status of the discovery process.

To stop the discovery process

Click STOP ALL SCANS.

No more runs are started and the discovery process attempts to stop all commands that are running. After the run has stopped, no more runs are performed until you start the discovery process.
Scanning IP addresses or ranges

You can start either an immediate or a scheduled discovery run by clicking the Add New Run link from the Discovery status page. You can perform the following for a run:

- Specify the IP range that is scanned during the normal discovery run.
- **Schedule a discovery run** to occur at any time and for a specified amount of time. For example, you might want to schedule a scan during off hours to avoid the risk of touching critical applications.
- Enable or disable individual scheduled discovery runs.
- Assign a company name to a discovery run.

For immediate runs, you can scan a specified address range right away, whether or not a regular discovery run is in progress. The specified addresses are scanned automatically and as soon as possible.

For general information about the discovery process, see the Discovery Engine.

**To scan IP addresses or ranges immediately**

1. On the discovery page, click **Add New Run**. For Consolidation appliances, click **Add New Local Run**.

2. Enter the information for the snapshot discovery run in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  
- IPv4 address (for example 192.168.1.100). Labelled v4.  
- IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4. |

**Note**

Scanning the following address types is not supported:  
- IPv6 link local addresses (prefix fe80::/64)  
- IPv6 multicast addresses (prefix ff00::/8)  
- IPv6 network prefix (for example fda8:7554:2721:a8b3::/64)  
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)
As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
</tbody>
</table>

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Level      | Select the level for the discovery run. This is one of:
|            | • **Sweep Scan** — This will do a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type.
|            | • **Full Discovery** — This will retrieve all the default info for hosts, and complete full inference. |
| Label      | Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown. |
| Company    | Select the company name to assign to the discovery run. This drop-down list is only displayed if multitenancy has been set up. If (No company) is displayed, or a company name that you were expecting is missing, refresh the list by clicking **Lookup Companies** on the CMDB Sync. See multitenancy for more information about this feature. |

3. Click **OK**.

The **Currently Processing Runs** tab is displayed with the new discovery run.

**To schedule a discovery run**

1. Click **Add New Run**. For Consolidation appliances, click **Add New Local Run**.
2. Select **Scheduled**.

The dialog displays Frequency, Start, and End menus.
3. Enter the information for the scheduled scan in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  
  • IPv4 address (for example 192.168.1.100). Labelled v4.  
  • IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4.  

**Note**

Scanning the following address types is not supported:  
• IPv6 link local addresses (prefix fe80::/64)  
• IPv6 multicast addresses (prefix ff00::/8)  
• IPv6 network prefix (for example fda8:7554:2721:a8b3::/64)  
• IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into **pills**, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.  
• To edit a pill, click the pill body and edit the text.  
• To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.  
• To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view. Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills** are not currently supported in Opera.

| Level       | Select the level for the discovery run. This is either:  
  • **Sweep Scan** — Performs a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type.  
  • **Full Discovery** — Retrieves all the default info for hosts, and complete full inference. |
Field name | Details
---|---
Label | Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown.
Company | Select the company name to assign to the discovery run. This drop-down list is only displayed if multitenancy has been set up. If (No company) is displayed, or a company name that you were expecting is missing, refresh the list by clicking **Lookup Companies** on the CMDB Sync. See multitenancy for more information about this feature.
Frequency | Select a frequency for the discovery run to be performed. This can be:
  - Weekly by days of week
  - Once per week
  - Monthly by day of month
  - Monthly by week of month.
Start | Based on the selected scan frequency, you are presented with different options to start scheduled scans. Discovery must be running at this time.
  - **Weekly by days of week**: You are provided with buttons for each day and drop down menus for the start time in hours and minutes. You can select one or more days from the day buttons. The selected buttons appear with a Yellow border.
  - **Once per week**: You are provided with drop down menus to select the day of the week and the start time in hours and minutes.
  - **Monthly by day of month**: You are provided with drop down menus to select the day of the month and the start time in hours and minutes.
  - **Monthly by week of month**: You are provided with drop down menus to the select the week, the day of the week, and the start time in hours and minutes.

For example, to start a scheduled weekly discovery run which starts on Friday at 19:30 hrs and continues until Saturday, you will do the following:
  - Select the scan frequency as **Weekly by days of week**.
  - Click on the **F** and **S** day buttons.
  - From the time drop down menu, select 19 hours and 30 minutes.
End | You can choose end a scheduled scan when it is completed by selecting **when completed**. Alternatively, use the available option to end the scan which is based on the selected scan frequency. If the duration of the end time expires before the scan has completed, then the scan is suspended until the next scheduled time that the scan occurs. The scan resumes from the point where it was previously suspended.
  - **Weekly by days of week**: You are provided with the drop down menus to select the end time in hours and minutes.
  - **Once per week**: You are provided with the drop down menus to select the day of the week and time to end the scan.
  - **Monthly by day of month**: You are provided with the menus to select the day of the month and time to end the scan.
  - **Monthly by week of month**: You are provided with the drop down menus to the select the number of days and the time within which the scan must end.

For example, to end a scheduled weekly discovery run which starts on Friday at 19:30 hrs and continues until 21 hours 30 minutes on Saturday (see the example for the Start field above), you will select 21 hours and 30 minutes from the time drop down menu.

4. **Click OK.**

The **Scheduled Runs** tab is displayed with the new scheduled discovery run.
To add another scan to the page, click **Add New Run**.
To delete any existing scheduled scans, select the entry and click **Delete**.

**To edit an existing scheduled run**
You can edit an existing scheduled run. If the run is currently in progress, it is automatically cancelled when you edit it.

1. From the **Scheduled Runs** tab of the Discovery Status page, click the scheduled run that you want to edit. If the run is currently in progress, it is cancelled if you make and apply any changes.
   The dialog displays all of the fields described in the previous table, all of which can be edited.
2. Make the required changes and click **OK**.

**To enable or disable a scheduled run**
You can enable or disable scheduled discovery runs in the following ways:

From the **Scheduled Runs** tab of the Discovery Status page, you can select individual, multiple, or all runs to enable or disable. If the run is currently in progress, it is cancelled if you make and apply any changes.

1. From the **Scheduled Runs** tab of the Discovery Status page, select the scheduled runs that you want to enable or disable. You can select all scheduled runs using the checkbox in the header row.
2. Click **Enable** to enable the runs, or **Disable** to disable the runs.

Scans enabled in this way do not run until the next scanning window, even if a scanning window is currently open.

From the **Edit Scheduled Run** dialog you can edit an existing scheduled run. If the run is currently in progress, it is automatically cancelled when you edit it.

1. From the **Scheduled Runs** tab of the Discovery Status page, click the scheduled run that you want to edit. If the run is currently in progress, it is cancelled if you make and apply any changes.
   The dialog displays all of the fields described in the previous table, all of which can be edited.
2. An **Enabled** checkbox is provided in the scheduling section. To disable the scheduled run, deselect the checkbox. To enable the scheduled run, select the checkbox.
3. Make the required changes and click **OK**.

Scans enabled in this way run in the current scanning window if it is open.
Excluding IP addresses and ranges from scanning

There might be some IP devices in your network that you either permanently do not want to be scanned, or want to exclude those from a scan at a scheduled time. For example, you might have some legacy applications which will only run on old hardware which might be regarded as fragile. You can add this device as a single IP exclude, or as an Exclude Range.

- To add an Exclude Range
- To edit an existing exclude range
- To enable or disable an exclude range

To add an Exclude Range

You can add the following types of exclude ranges:

- Permanent
- Scheduled

To add a permanent exclude range

1. On the Discovery Status page, click Add New Exclude.
2. Enter the information for the exclude range in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  
- IPv4 address (for example 192.168.1.100). Labelled v4.  
- IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4.  
As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.  
Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.  
Warning: You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.  
- To edit a pill, click the pill body and edit the text.  
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.  
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.  
Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.  
Pills are not currently supported in Opera. |
| Label      | Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown. |
| Description| A free text description of the exclude range. |
3. Click **OK**.  
The **Excluded Ranges** tab is displayed with the new exclusion.

**To add a scheduled exclude range**

1. On the Discovery Status page, click **Add New Exclude**.  
2. Enter the information for the exclude range in the fields.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Range      | Enter IP address information in one of the following formats:  
  - IPv4 address (for example 192.168.1.100). **Labelled v4**.  
  - IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). **Labelled v4**. |

**Large scheduled exclude ranges**

If you specify a scheduled exclude range which blocks 65535 IP addresses or more of a scan range, new IP addresses from that scan are not added to the scan queue until at least one of the blocked IP addresses is allowed to scan.

As you enter text, the UI divides it into * pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.  
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.  
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.  

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Enter a label for the discovery run. Where the discovery run is referred to in the UI, it is this label that is shown.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the exclude range.</td>
</tr>
<tr>
<td>Frequency</td>
<td>Add text</td>
</tr>
</tbody>
</table>
| Start      | Based on the selected frequency of the scheduled exclude, you are presented with the following start options:  
  - **Weekly by days of week:** You are provided with buttons for each day and drop down menus for the start time in hours and minutes. You can select one or more days from the day buttons.  
  - **Once per week:** You are provided with drop down menus to select the day of the week and the start time in hours and minutes.  
  - **Monthly by day of month:** You are provided with drop down menus to select the day of the month and the start time in hours and minutes.  
  - **Monthly by week of month:** You are provided with drop down menus to select the week, the day of the week, and the start time in hours and minutes. |
For example, to start a scheduled weekly exclude which starts on Friday at 19:30 hrs and continues until Saturday, you will do the following:

a. Select the exclude frequency as Weekly by days of week.

b. Click on the F and S day buttons.

c. From the time drop down menu, select 19 hours and 30 minutes.

End

Based on the selected frequency of the scheduled exclude, you are presented with the following end options:

- **Weekly by days of week**: You are provided with the drop down menus to select the end time in hours and minutes.
- **Once per week**: You are provided with the drop down menus to select the day of the week and time to end a scheduled exclude.
- **Monthly by day of month**: You are provided with the menus to select the day of the month and time to end a scheduled exclude.
- **Monthly by week of month**: You are provided with the drop down menus to the select the number of days and the time within which a scheduled exclude must end.

For example, to end a scheduled weekly exclude which starts on Friday at 19:30 hrs and continues until 21 hours 30 minutes on Saturday (see the example for the Start field above), you will select 21 hours and 30 minutes from the time drop down menu.

### To edit an existing exclude range

You can edit an existing exclude range. If the exclude range is currently in progress, it is automatically cancelled when you edit it.

1. From the **Exclude Ranges** tab of the Discovery Status page, click the exclude range that you want to edit.
   
   All of the fields are editable.

2. Make the required changes and click **OK**.

### To enable or disable an exclude range

You can enable or disable an existing exclude range, whether it is permanent or scheduled.

From the **Exclude Ranges** tab of the Discovery Status page, select individual, multiple, or all exclude ranges to enable or disable. If the exclude range is currently in progress, it is cancelled if you make and apply any changes.

### Stalled discovery runs

When a discovery run is scheduled for a particular scan range, the scheduled time is called its scan window. Scheduled discovery of the scan range can only take place in that open scan window.

- To identify stalled discovery runs
- To restart or continue
- To identify locked and on hold runs in the UI
To view the status of the Reasoning service by using the tw_reasoning_status utility

To identify stalled discovery runs

Some discovery runs can appear to be stalled. This occurs in one of two situations:

- A discovery run has not completed within its scheduled scan window. This scan is considered on hold. The run continues or restarts at the next scan window.
- Discovering an endpoint requires additional discovery on another endpoint which is not currently in an open scan window. This scan is blocked.

When the scan window for the other endpoint opens, the endpoint is scanned, irrespective of whether the first endpoint is in an open scan window or not.

To restart or continue

At the next scheduled scan window, any runs that are on hold or blocked will continue or restart, depending on whether all endpoints have been started or not. The following happens:

- Scan fails to complete in scan window and all endpoints have not been started. Scan continues at next window.
- Scan fails to complete in scan window and all endpoints have been started. Scan restarts at next window.

To identify locked and on hold runs in the UI

The Currently Processing Runs tab of the Discovery Status window displays a red (on hold) notice if a run is on hold and a red (blocked) notice if a run is blocked.

Blocked runs

Click the (blocked) notice. A dialog box is displayed showing why the discovery run is blocked.

The blocked endpoint is shown and the reason for it being blocked is given. In the preceding example, the blocked endpoint is 137.72.94.27 and it is blocked because the DiscoveryRunCommand.DiscoveryRunCommand pattern is attempting to access another endpoint (137.72.94.219) which is currently outside a scan window.

On hold runs

Click anywhere in the row of the on hold run to display the Discovery Run page for that run. To check the scanning window:

1. From the Discovery Status page click the Scheduled Runs tab.
2. The timing information for each scheduled run is shown in a table.

If you want to edit the discovery run, click its entry in the table to display the Edit an Existing Run dialog box.
To view the status of the Reasoning service by using the `tw_reasoning_status` utility

The `tw_reasoningstatus` utility enables you to view the status of the Reasoning service. Typically this utility is used by Customer Support as a troubleshooting tool for investigating possible problems.

⚠️ **Automatic use of `tw_reasoningstatus`**

Reasoning runs the same status check automatically every 15 minutes and outputs the results in the `tw_svc_reasoning.log` file.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_reasoningstatus [options]
```

where `options` are either the command described in the following table or the standard, inherited options detailed in Using command line utilities.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--waiting, -w</code></td>
<td>Lists information for all endpoints which are on hold waiting for information from the discovery of a different endpoint.</td>
</tr>
<tr>
<td><code>--waiting-full</code></td>
<td>Expands the information provided by the <code>--waiting</code> option to include information on all endpoints being held waiting for discovery. This option is ignored if <code>--waiting</code> is not specified.</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specifies the name of the BMC Atrium Discovery user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**Example of viewing the status of the Reasoning service**

1. On the command line, type the following:
   
   ```
   $TIDEWAY/bin/tw_reasoningstatus --username joe
   ```

2. If you do not provide a password, you are prompted for one.
   After providing a password, a status is displayed that includes information about engine status, pool state, queue length, and so forth. The output is saved in the `tw_svc_reasoning.log` file.

**Edge connectivity**

Edge connectivity refers to the manner in which hosts are connected to edge switches.

- How edge connectivity works
- To enable or disable edge connectivity
How edge connectivity works

Edge connectivity discovery occurs when a discovery run takes place. When a switch is discovered, it is queried for port information. The SNMP request `{getNetworkConnectionList}` method gathers information on all ports found on the switch, as well as a list of neighbor MAC addresses found for each port.

When a host is discovered, its MAC address or addresses are determined. Reasoning uses the MAC addresses to determine whether there is a connection between that host and any previously scanned switches and, if so, creates a new one. If a connection already exists, then it is confirmed. If no connection is found, any existing connection is deleted.

- When a host is discovered, an old connection might be deleted, and a connection to a switch might be made or confirmed.
- When a switch is discovered, an old connection might be deleted. No new connections are made.

⚠️ **Note**

Connections are only made when a host is discovered, and when the switch that it is connected to has already been scanned. Both ends of the connection are required before the connection can be confirmed. Consequently, connection information might not be complete until two scans have taken place.

The provenance of a connection is the provenance of the host or the switch, whichever was discovered last.

To enable or disable edge connectivity

Edge connectivity is enabled by default in a fresh installation of BMC Atrium Discovery version 8.3. See upgrades for information on the behavior of upgraded appliances.

If you do not intend to use the connection information obtained by edge connectivity, you can disable it on the Discovery Configuration page. Disabling edge connectivity avoids any additional load put onto switches in your infrastructure. To do this, an additional option Discover neighbor information when scanning network devices is provided in Discovery Configuration. This option controls whether the information on neighboring ports is obtained.

- Select Yes to retrieve neighbor MAC addresses (enable edge connectivity)
- Select No if you do not want to retrieve neighbor MAC addresses (disable edge connectivity)

If you do disable edge connectivity, all existing connections are deleted at the time of the next scan.
## Discovery targets

The following topics provide information about discovery targets and instructions for their configuration:

- Credentials
- Discovering UNIX, Linux and related OS based hosts
- Discovering Windows Hosts
- Discovering SNMP devices
- Discovering Load Balancers
- Discovering Management Controllers
- Discovering ESX and ESXi hosts
- Discovering storage
- Discovering database content
- Discovering WebLogic
- Discovering Tomcat
- Discovering mainframe computers
- Discovering IBMi
- Discovering WebSphere
- Discovering IBM Power Systems
- Discovering Cisco Nexus switches

### Credentials

During discovery, the BMC Atrium Discovery system attempts to access host systems to obtain details of processes running. Credentials including IDs and passwords, and credential-like entities (Windows proxies and SNMP credentials) for different access methods, can be stored on the system to allow the required level of access. You can set up the following:

- **Login credentials** (user IDs and passwords) for interactive log-in to different host systems.
- **Windows proxies** used to discover Windows systems.
- **SNMP credentials** used on particular host systems. SNMP queries are only tried if an attempted login fails and if the SNMP port (UDP 161) is open on the target host.
- **vCenter credentials** used to discover VMware ESX and ESXi hosts by querying the vCenter server, including the management system vCenter credentials that are used to log into the vCenter. This view of the vCenter credentials is populated when patterns containing vCenter queries are activated.
- **vSphere credentials** used to discover VMware ESX and ESXi hosts, including the management system vSphere credentials that are used to log into the vSphere API. This view of the vSphere credentials is populated when patterns containing vSphere queries are activated.
- **WBEM credentials** used to discover storage systems managed using WBEM.
XML API credentials used to discover devices with XML based APIs such as HP iLO devices.

Database credentials used to log into databases and query their content.

Middleware credentials used to query middleware such as web and application servers.

Mainframe credentials used to discover mainframe computers.

You can also run and view the progress of Device Credential Tests, as well as see the credentials that match the specific IP address or IP address range.

About credential storage

BMC Atrium Discovery stores all passwords in a secure credential vault. Secure credential vaults are generated using public and private key pair when the appliance is commissioned. The contents of the vault is encrypted, and can be secured using a passphrase.

User accounts on the target system

When creating a user account (the account that BMC Atrium Discovery logs into to discover a host) on a target host, you should ensure that the full path to the shell is entered in the user's profile. If you do not do this, the credentials will be considered invalid.

For example: `SHELL=/bin/sh`

Shell support

BMC Atrium Discovery is tested to work with Bourne and Bourne-compatible shells. Support for other shells such as the Korn shell is best effort only. The product has been sporadically tested and might work but with known issues and we might not fix bugs that affect these shells.

Testing credentials

This topic provides information about testing discovery credentials for both devices and mainframes:

- Accessing credential tests
- Managing credential tests
- Testing credentials for an arbitrary IP Address
- Testing existing login credentials from the Host page

Accessing credential tests

You can test discovery credentials using the Credential Tests page or Mainframe Credential Tests page. To access these pages:

1. From the secondary navigation bar on the Discovery tab, click Credentials.
2. Open the necessary section:
   - For Credential Tests, click Devices.
- For Mainframe Credential Tests, click **Management System**.

3. Click **Credential Tests** or **Mainframe Credential Tests**.

The (mainframe) credential tests page is displayed:

Example of the device credential tests:

![Example of the device credential tests](image)

Example of the mainframe credential tests:

![Example of the mainframe credential tests](image)

### Managing credential tests

The Credential Tests page and Mainframe Credential Tests page display the credential tests that are currently in progress and the tests run in the last ten minutes. You can delete completed tests, or run them again. However, you cannot cancel tests that are currently in progress. You can also test whether credentials exist for any IP address on your network.

- To delete a credential test, in the **Actions** list of that test, click **Delete**.
- To run a credential test again, in the **Actions** list of that test, click **Retry**.

### Testing credentials for an arbitrary IP Address

You can test whether BMC Atrium Discovery can access any device or mainframe IP address on your network. You do not need to have previously scanned the IP address. This way you can see whether any of your existing device/mainframe credentials or Windows proxies allows you to access this IP.

To test device/mainframe credentials for an arbitrary IP Address, perform the following:

1. From the (Mainframe) Credential Test page, click **Test IP Access...** at the bottom right. The **Test Device Credentials** or **Test Management System Credentials** dialog box is displayed.
2. Enter the IP address that you would like to test the credentials with.
3. Click **Test**.
   While the test runs, the test state is shown as "In progress". When the test completes, the page is refreshed and the result state is updated.

4. You are shown whether the test ultimately resulted in a success or a failure. By clicking through the link in a State column, you can view the results of the credential test in more detail.

![Test result for arbitrary IP address.](image)

5. For ssh credentials, a verbose session log is displayed.
   To view the session log, click the show session log link.

The credential test results page shows you all the credentials, access methods and Windows proxies that were tried during the test, and which failed (if any). As would happen during a normal discovery, BMC Atrium Discovery continues until either a successful access is obtained, or it runs out of methods to try.

**Testing existing login credentials from the Host page**

You can test existing login credentials when viewing a host's details.

1. From a Host page, click **Actions**.

![Testing credentials from the host page.](image)

2. Click **Test Discovery Credentials**.
   The results are displayed; this can take a few minutes.
If you do not have the correct credentials, a message will be displayed to this effect. If this occurs you must exit the test and select another host or configure an appropriate credential.

To identify matching credentials

1. From the **Credentials > Devices > Hosts** page, click **Show Matching Credentials**.
2. Enter the IP address that you want to check in one of the following formats:
   - IPv4 address (for example 192.168.1.100). Labelled **v4**.
   As you enter text, the UI formats it as a *pill*, a discrete editable unit, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labelled with a question mark. You can only enter one IP address in this dialog.
   - To edit the pill, click the pill body and edit the text.
   - To delete the pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
3. Click **Search**.
   Any credential that has an IP range matching the specified IP address is highlighted in yellow, and a banner summarizing the results is displayed just below the page heading. The matching does not include **disabled** credentials.

If no matching credentials are found this is stated in the banner.

The **Show matching credentials** button is available on the following credential pages:

- Hosts
- SNMP
- vCenter
- vSphere
- Mainframe

**Discovering UNIX, Linux and related OS based hosts**

The following topics provide information and instructions for discovering UNIX, Linux and related OS based hosts:
• Configuring host login credentials
• User privileges and information access for UNIX and related operating systems
• Standalone UNIX scanning

Configuring host login credentials

The preferred method of accessing remote hosts using BMC Atrium Discovery is by remote login. You can set up different login credentials to use on different computers, by individual IP address or a range of addresses. The following access methods are available:

- ssh
- telnet
- rlogin
- windows

You can set up several access methods and define the order in which they are to be attempted. Each access method is attempted until a working credential is found or the list is exhausted.

For each host that is successfully logged into, the successful access method is recorded. On subsequent scans, the first access method attempted is the one that succeeded for that host on the previous attempt, as long as the appropriate option is selected in the Discovery Configuration page.

If an access login method is disabled (for example, telnet) and that method is recorded as the last successful login method, it is tried again on a subsequent scan. If it fails on that scan then that method will not be tried again until it is re-enabled.

An access method is only attempted if it is seen to be available (for example, SSH access will only be attempted if the SSH port is open).

The following pages describe how to configure and view host login credentials:

• Adding host login credentials
• Viewing host login credentials
• Managing the credential vault
• Using SSH keys

Adding host login credentials

To add host login credentials

From the login credentials page, click Add.

Set up the login credentials as follows:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching criteria</td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td>Select &quot;Match All&quot; to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:</td>
</tr>
<tr>
<td></td>
<td>• IPv4 address: for example 192.168.1.100.</td>
</tr>
<tr>
<td></td>
<td>• IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.</td>
</tr>
<tr>
<td></td>
<td>• IPv6 network prefix: for example fda8:7554:2721:a8b3::/64.</td>
</tr>
</tbody>
</table>

**Note**

The following address types cannot be specified

• IPv6 link local addresses (prefix fe80::/64)
• IPv6 multicast addresses (prefix ff00::/8)
• IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Click here for more information on using the pill UI.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning**

You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

• To edit a pill, click the pill body and edit the text.
• To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
• To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Information**

Pills are not currently supported in Opera.

Matching criteria

Select "Match All" to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:

• IPv4 address: for example 192.168.1.100.
• IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.
• IPv6 network prefix: for example fda8:7554:2721:a8b3::/64.
The following address types cannot be specified

- IPv6 link local addresses (prefix fe80::/64)
- IPv6 multicast addresses (prefix ff00::/8)
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Click here for more information on using the pill UI. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

Warning
You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

Information
Pills are not currently supported in Opera.

Enabled
A check box to define whether or not the credential is enabled. You can also enable or disable existing credentials on the Credentials page.

Username
Username used to log in to hosts identified by the key. If this is a Windows credential that will be used by a pre-8.2 Windows credential proxy, ensure that you add a localhost prefix to the user name (for example, localhost\Administrator).

Password
Enter the password into the password entry field; the password text is not echoed to the screen.

Note
In the Edit Login Credential page, this field is displayed as Set Password. The existing password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Now enter the new password.

Description
A free-text description of this login credential.

Access Methods
Choose the access methods to be attempted for any host identified by the key by selecting them and moving them to the right-hand (enabled) list box using the right arrow button. By default, all access methods are placed in this box, that is, they are all enabled. You can also change the order in which the access methods are attempted by selecting them and moving them up or down with the up or down arrow buttons.

If you want to create a session log, select Enabled. This logs all communication between the BMC Atrium Discovery appliance and a host and should only be used for diagnosing discovery problems with that host. There is currently no option for recording a session log for Windows hosts.

You can also change the order in which the access methods are attempted by selecting them and moving them up or down with the up or down arrow buttons.

### Field Name | Details
--- | ---
**Choose the access methods to be attempted for any host identified by the key by selecting them and moving them to the right-hand (enabled) list box using the right arrow button. By default, all access methods are placed in this box, that is, they are all enabled. You can also change the order in which the access methods are attempted by selecting them and moving them up or down with the up or down arrow buttons.**

| **Session Logging** | If you want to create a session log, select Enabled. This logs all communication between the BMC Atrium Discovery appliance and a host and should only be used for diagnosing discovery problems with that host. There is currently no option for recording a session log for Windows hosts. |
| **Prompt** | A regular expression to define valid prompt characters expected. |
| **SU** | To use the su command to change to the root or any other user, select Switch User. Enter the user to change to, and the corresponding password. The password text is not echoed to the screen. |
| **Timeout** | Enter a timeout period (in seconds) for a session. This timeout includes the credential handshaking (see also the Session Login Timeout). This timeout is used to control sessions. The default is 180 seconds. In general, it is not used to limit the time to scan devices. More than one session can be used to scan one device. For this reason, a scan can take more time than this timeout. A typical consequence of this timeout (for example, when the execution of the platform script for getInterfaceList takes longer than this timeout) is that the scan will fail with a script failure (error message Connection timed out). |
| **Force Subshell** | To force the session to open a Bourne (/bin/sh) subshell, if the default login shell is a C shell (/bin/csh /bin/tcsh), select Yes. This enables you to cater for machines using non-standard shells. |
| **SSH Key** | Specify an existing SSH key which you already have deployed in your organization. Click Browse to locate the private key and click Open to select it. Enter the passphrase in the passphrase field. When you click Apply to save the credential, the key and passphrase are validated. When you upload the private key to the appliance it is strongly recommended that you protect the vault with a passphrase. See Using SSH keys for more detailed information. |
| **SSH Authentication** | To use an SSH key or password, select Key or Password. If you have not configured an SSH key, Key is dimmed. |
| **Custom SSH Port** | If the host for which this credential is intended is configured to listen for SSH connections on a non-standard port, enter this here. To do this, select the Enable custom ssh port? check box and enter the port number in the entry field. If you add a port here, it is automatically added to the TCP ports to use for initial scan. For more information, see TCP and UDP ports to use for initial scan. |
| **Custom Telnet Port** | If the host for which this credential is intended is configured to listen for Telnet connections on a non-standard port, enter this here. To do this, select the Enable custom telnet port? check box and enter the port number in the entry field. If you add a port here, it is automatically added to the TCP ports to use for initial scan. For more information, see TCP and UDP ports to use for initial scan. |

1. To add the credentials, click Apply. Repeat Steps 1 through 3 for all the credentials you want to add.

To test host login credentials
When you have added the credentials, you can test them.

1. On the login credentials page, click Actions for the login credential.
2. Select Test.
3. To test the credential, enter a single IP address in the IP Address field. For example, 137.72.93.222.
4. Click Test.
   You can perform other credential tests from the Credential Tests page.

To edit host login credentials

1. From the login credentials page, click Actions for the credential.
2. Select Edit.
3. On the Edit Login Credential page, edit the host login credential fields.
   The fields in this page are the same as the fields in the Create Login Credential page. For more information about the fields, see the field-details table for Setting up host login credentials.
4. To add the edited credentials, click Apply.

Fixing invalid credentials

Where a credential was specified using a regex or wildcard in a previous version and has been brought into BMC Atrium Discovery version 9.0, it is marked as invalid. Before the credential can be used, the range must be corrected in the Matching Criteria field of the Edit Credential dialog. You cannot save an invalid credential.

Here are some examples of how to convert them:

<table>
<thead>
<tr>
<th>Old format</th>
<th>New format</th>
</tr>
</thead>
<tbody>
<tr>
<td>137.72.(94</td>
<td>95).*</td>
</tr>
<tr>
<td></td>
<td>137.72.94.<em>, 137.72.95.</em></td>
</tr>
</tbody>
</table>

To view login credentials

1. From the secondary navigation bar on the Discovery tab, click Credentials.
2. Click Devices.
   The following screen illustrates an example of the host login credentials page:

⚠️ At the top of the host login credentials page, the system displays the total number of host login credentials and the number of host login credentials by each access method.
The credentials are checked in sequence, and the first matching entry is used. After a working credential is found, further credentials are not checked. If you want to reorder login credentials, drag the credential to the required position in the list.

The credentials are shown in color-coded boxes. The colors represent the level of login success achieved with that credential:

- **Green** — 100% success rate.
- **Yellow** — partial success.
- **Blue** — the credential has never been used.
- **Red** — 0% success rate.

Where a credential is shown in a filled red box, it is an invalid credential from a previous BMC Atrium Discovery version. Previous versions have permitted range specifications using regular expressions. Before the credential can be used, the range must be corrected in the Matching Criteria field of the [Edit Credential] dialog.

The following information is shown for each credential:

<table>
<thead>
<tr>
<th>Credential link</th>
<th>This is the first part of the heading link for the credential. The range of IP addresses on which this credential is intended to be used.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>This is the second part of the heading link for the credential. The user name used for this credential.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td>Usage</td>
<td>A summary of the success rate when the credential has been used, information about failures, and links to DiscoveryAccesses, credential lists and other useful diagnostic pages.</td>
</tr>
<tr>
<td>Options</td>
<td>Additional options used with this credential. With the exception of [No Password (use ssh key exchange)], the options are those selected from the Options section when the credential is set up. The [No Password (use ssh key exchange)] option is selected by not entering a password. For information about these, see the Field Name-Details table for [Adding host login credentials].</td>
</tr>
</tbody>
</table>

### Actions

- **Edit** — To edit the credential, select Edit. See [Adding host login credentials] for information about the fields and settings available from the Edit Login Credential page.
- **Disable** — To disable a credential, select Disable. The credential is a marked as disabled in the credential list. When a credential is disabled, this option is replaced with an Enable option. To enable the credential, click Enable.
- **Delete** — Delete the credential.
- **Test** — Select this to test the credential. See [Adding host login credentials] and [To test host login credentials] for more information.
- **Move to top** — Moves the credential to the top of the list.
- **Move to bottom** — Moves the credential to the bottom of the list.

### Managing the credential vault

The credentials used to log in to discovery targets, synchronize to the CMDB, and export data using adapters are stored in a vault that is encrypted with a default passphrase when the appliance is built. The vault provides a secure mechanism for storing credential information. Only users with
Discovery or Administration privileges have read/write access to the vault, with read access limited to non-sensitive information only (passwords can never be seen in the UI or at the command line). The content of the vault is secured using 256 bit AES encryption in CBC mode.

For further details, see Information security.

The credential vault can be open or closed. If no passphrase is set, the vault is opened automatically when Discovery starts. If a passphrase has been set, you will be prompted to enter it before Discovery can begin. While the Vault is open, BMC Atrium Discovery can use the credentials stored in it to access devices.

When Discovery is stopped, the vault is automatically closed if a passphrase is set. You can close the Vault while Discovery is in progress. This will prevent access to further devices during the current Discovery runs.

Whenever a credential is added, removed, or changed, the vault is backed up. No more than two copies of the vault are held as back ups. When the vault passphrase is added, changed, or removed, all backups are deleted, ensuring that no backups of potentially less secure vaults are retained on the system.

To manage the credential vault
From the Discovery section of the Administration tab, select Vault Management. From this page you can open or close the credential vault and specify a passphrase to secure it. You can also change the passphrase or remove it.

Setting a passphrase
To set a passphrase:

1. Enter the new passphrase in the New Passphrase field.
2. Repeat it in the verify New Passphrase field.
3. Click Set Passphrase.
   The passphrase is now set.

Changing a passphrase
To change a passphrase:

1. Enter the new passphrase in the New Passphrase field.
2. Repeat it in the Verify New Passphrase field.
3. Click Change Passphrase.
   The passphrase is now changed.

⚠️ Setting or changing a passphrase does not change whether the vault is open or closed.

Clearing a passphrase
To clear a passphrase:

1. Enter the current passphrase in the **Current Passphrase** field.
2. Click **Clear Passphrase**.
   The passphrase is now cleared.

**Opening the credential vault**
To open a closed credential vault:
Enter the passphrase and click **Open the Vault**.
You are requested to confirm the operation.

You can also open the credential vault from the Discovery Status page. When Discovery is not running and the vault is closed, a Passphrase entry box is displayed above START LOCAL SCANS.

**Closing the credential vault**
To close the vault, it must be open and have the passphrase set:
Enter the passphrase and click **Close the Vault**.
You are requested to confirm the operation.

You can also close the vault from the Discovery Status page. When Discovery is running and a passphrase is set, stopping Discovery also closes the vault.

**Using SSH keys**
You can attach an SSH key to any credential using the SSH access method.

> BMC Atrium Discovery supports only RSA2 and DSA private keys for certificate-based authentication. For hosts that only support SSH v1, you must use credentials for authentication.

When using SSH keys, the appliance must identify itself to discovery targets so must use the <code>private</code> key. It is strongly recommended that you protect the private key with a strong passphrase. When they are uploaded to the appliance, the key and the passphrase are stored in the credential vault.

If the attempted login is unsuccessful using the SSH key, the credential falls back and attempts to login using the configured username and password.

It is important to configure a user name and password even when an SSH key is to be used. When privileged command execution is required, that password is used in the command, for example <code>sudo password command</code>.
After the key is stored in the credential vault, it is encrypted and cannot be recovered from the vault. You are *strongly* recommended to keep copies of private keys in secure storage according to your local security guidelines.

User privileges and information access for UNIX and related operating systems

This page shows the information categories discovered, the script used, and whether privileged access is required. Optional commands such as `lsof` are assumed to have been installed in all cases.

All methods which are marked with an asterisk (*) must succeed for a host to be created. For a method to succeed, it must return some useful data. Where multiple scripts are listed against such methods, one of the scripts must succeed. For simple methods such as `getDeviceInfo` and `getHostInfo` this means the result contains expected attributes such as hostname, OS information and so on. For more complex methods, such as `getNetworkInterfaces`, one of the defined scripts must return at least one item, an interface. Additional data such as speed or duplex is not required.

**AIX**

**Shell scripts**

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo’</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td>hbainfo</td>
<td></td>
</tr>
<tr>
<td>getHostInfo’</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses’</td>
<td>lsdev_lscfg_mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td>netstat_link_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifdetails</td>
<td></td>
</tr>
<tr>
<td></td>
<td>netstat_link_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>lslpp</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Script</td>
<td>Privileges required</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>--------------</td>
<td>---------------------</td>
</tr>
<tr>
<td><code>getProcessToConnectionMapping</code></td>
<td><code>lsnf-i</code></td>
<td></td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>getDeviceInfo</code></td>
<td><code>SNMPv2-MIB::sysDescr.0</code></td>
<td><code>1.3.6.1.2.1.1.1.0</code></td>
</tr>
<tr>
<td></td>
<td><code>SNMPv2-MIB::sysName.0</code></td>
<td><code>1.3.6.1.2.1.1.5.0</code></td>
</tr>
<tr>
<td><code>getHostInfo</code></td>
<td><code>HOST-RESOURCES-MIB::hrSystemUptime.0</code></td>
<td><code>1.3.6.1.2.1.25.1.1.0</code></td>
</tr>
<tr>
<td></td>
<td><code>HOST-RESOURCES-MIB::hrMemorySize.0</code></td>
<td><code>1.3.6.1.2.1.25.2.2.0</code></td>
</tr>
<tr>
<td><code>getIPAddresses</code></td>
<td><code>IF-MIB::ifEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ifDescr, ifType, ifOperStatus ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IP-MIB::ipAddressEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IP-MIB::ipAddrEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipv6AddrAddress, ipv6AddrPrefixLength ]</code></td>
<td></td>
</tr>
<tr>
<td><code>getMACAddresses</code></td>
<td><code>IF-MIB::ifEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ifDescr, ifType, ifPhysAddress, ifOperStatus ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IP-MIB::ipNetToPhysicalEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipNetToPhysicalAddress, ipNetToPhysicalType ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IP-MIB::ipNetToMediaEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipNetToMediaAddress, ipNetToMediaType ]</code></td>
<td></td>
</tr>
<tr>
<td><code>getNetworkConnectionList</code></td>
<td><code>TCP-MIB::tcpConnectionEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>TCP-MIB::tcpListenerEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>UDP-MIB::udpEndpointEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess ]</code></td>
<td></td>
</tr>
<tr>
<td><code>getNetworkInterfaces</code></td>
<td><code>TCP-MIB::tcpConnEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ tcpConnState, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IPv6-TCP-MIB::ipv6TcpConnEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>UDP-MIB::udpConnEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ udpLocalAddress, udpLocalPort ]</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>IPv6-UDP-MIB::ipv6UdpEntry</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>[ ipv6UdpLocalAddress, ipv6UdpLocalPort ]</code></td>
<td></td>
</tr>
</tbody>
</table>
### Table of MIB Values and OID

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF-MIB::ifEntry</td>
<td>[ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td>IF-MIB::ifXEntry</td>
<td>[ifAlias, ifName, ifHighSpeed]</td>
<td>1.3.6.1.2.1.2.1.26.2.1.1</td>
</tr>
<tr>
<td>MAU-MIB::ifMauEntry</td>
<td>[ifMauIndex, ifMauType, ifMauAutoNegSupported]</td>
<td>1.3.6.1.2.1.2.1.31.1.1.1</td>
</tr>
<tr>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>[dot3StatsDuplexStatus]</td>
<td>1.3.6.1.2.1.1.1.0.7.2.1</td>
</tr>
<tr>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>[ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress, ipNetToPhysicalType]</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>[ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
</tbody>
</table>

### FreeBSD

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td>Privileges</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>pkg_info</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td></td>
</tr>
</tbody>
</table>
## SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysDescr.0, SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0, HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddress</td>
<td>IF-MIB::ifEntry [ifDescr, ifType, ifOperStatus]</td>
<td>1.3.6.1.2.1.1.4.1.34.1</td>
</tr>
<tr>
<td></td>
<td>[ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix]</td>
<td>[2, 3, 4, 5]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrEntry [ipAddrAddr, ipAddrIfIndex, ipAddrNetMask]</td>
<td>1.3.6.1.2.1.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry [ipv6AddrAddress, ipv6AddrPrefix]</td>
<td>1.3.6.1.2.1.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry [ifDescr, ifType, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToLogicalEntry [ipNetToLogicalPhysAddress, ipNetToLogicalPhysAddress]</td>
<td>[2, 3, 6, 8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry [ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>1.3.6.1.2.1.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry [tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess]</td>
<td>1.3.6.1.2.1.6.1.19.1</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry [tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess]</td>
<td>[2, 3, 5, 6, 7, 8]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry [udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess]</td>
<td>1.3.6.1.2.1.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 4]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 8]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpConnEntry [tcpConnState, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>IPv6-TCP-MIB::ipv6TcpConnEntry [ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState]</td>
<td>[1, 2, 3, 4, 5]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry [udpLocalAddress, udpLocalPort]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2, 3, 4, 6]</td>
</tr>
<tr>
<td></td>
<td>IPv6-UDP-MIB::ipv6UdpConnEntry [ipv6UdpLocalAddress, ipv6UdpLocalPort]</td>
<td>1.3.6.1.2.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry [ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry [ifAlias, ifName, ifHighSpeed]</td>
<td>[1, 2, 3, 5, 6, 8]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry [ifMauIndex, ifMauType, ifMauAutoNegSupported]</td>
<td>1.3.6.1.2.1.31.1.1.1</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry [dot3StatsStatus]</td>
<td>[1, .15, .18]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.1.26.1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.1.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[19]</td>
</tr>
</tbody>
</table>
### Method | MIB Values | OID
--- | --- | ---
**getDeviceInfo** | [ ipNetToPhysicalIfIndex, ipNetToPhysicalAddress, ipNetToPhysicalType ]
        | IP-MIB::ipNetToMediaEntry
        | [ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ] | 1.3.6.1.2.1.4.35.1 [.1, .4, .6 ]
        | | 1.3.6.1.2.1.4.22.1 [.1, .2, .4 ]

### getPackageList
**Method** | HOST-RESOURCES-MIB::hrSWInstalledTable
| [ hrSWInstalledName ] | 1.3.6.1.2.1.25.6.3.1 [.2 ]

### getProcessList
**Method** | HOST-RESOURCES-MIB::hrSWRunTable
| [ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ] | 1.3.6.1.2.1.25.4.2.1 [.1, .2, .4, .5 ]

### HPUX

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>bdf</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td>fcmutil</td>
<td>Privileges</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>lanscan_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>interface_commands</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>swlist</td>
<td>Privileges</td>
</tr>
<tr>
<td>getPatchList</td>
<td>patch_list</td>
<td>Privileges</td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td>Privileges</td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
</table>
| getDeviceInfo | SNMPv2-MIB::sysDescr.0
        | SNMPv2-MIB::sysName.0 | 1.3.6.1.2.1.1.1.0
<pre><code>    | | 1.3.6.1.2.1.1.5.0 |
</code></pre>
<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getHostInfo*</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifOperStatus ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[.2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddEntry</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask ]</td>
<td>[.1, .2, .3]</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6AddrAddress, ipv6AddrPfxLength ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td>getMACAddresses*</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifPhysAddress, ifOperStatus ]</td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIpAddress, ipNetToPhysicalType ]</td>
<td>[.4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.2, .4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort,</td>
<td>[.2, .3, .5, .6, .7, .8]</td>
</tr>
<tr>
<td></td>
<td>tcpConnectionRemAddress, tcpConnectionRemPort,</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>tcpConnectionState, tcpConnectionProcess ]</td>
<td>[.2, .3, .4]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>[ udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess ]</td>
<td>[.1, .2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>IPv6-TCP-MIB::ipv6TcpConnEntry</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort ]</td>
<td>[.1, .2, .3, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>IPv6-TCP-MIB::ipv6TcpConnRemAddress, ipv6TcpConnRemPort,</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td>ipv6TcpConnState, ipv6TcpConnRemPort ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IPv6-UDP-MIB::ipv6UdpConnEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6UdpLocalAddress, ipv6UdpLocalPort ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ]</td>
<td>[.1, .2, .3, .5, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.31.1.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifAlias, ifName, ifHighSpeed ]</td>
<td>[.1, .15, .18]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.26.2.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td>[.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td>[.19]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalIpAddress, ipNetToPhysicalType ]</td>
<td>[.1, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.1, .2, .4]</td>
</tr>
</tbody>
</table>
### IRIX

**Shell scripts**

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileInfo</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>netstat_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifconfig_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>showprods</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td>Privileges</td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifOperStatus ]</td>
<td>[ .2, .3, .8 ]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry</td>
<td>1.3.6.1.2.1.2.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[ .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::iifEntry</td>
<td>1.3.6.1.2.1.4.20.1 [ .1, .2, .3 ]</td>
</tr>
<tr>
<td></td>
<td>[ iifDescr, iifType, iifPhysAddress, iifOperStatus ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1 [ .4, .6 ]</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1 [ .2, .4 ]</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1 [ .2, .3, .5, .6, .7, .8 ]</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.6.20.1 [ .2, .3, .4 ]</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry</td>
<td>1.3.6.1.2.1.7.7.1 [ .2, .3, .8 ]</td>
</tr>
<tr>
<td></td>
<td>[ udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess ]</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::iifEntry</td>
<td>1.3.6.1.2.1.6.13.1 [ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>[ iifIndex, iifDescr, iifType, iifSpeed, iifPhysAddress, iifOperStatus ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IF-MIB::iifXEntry</td>
<td>1.3.6.1.2.1.25.6.3.1 [ .2 ]</td>
</tr>
<tr>
<td></td>
<td>[ iifAlias, iifName, iifHighSpeed ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.2.2.2.1 [ .1, .2, .3, .5, .6, .8 ]</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.2.26.2.1.1 [ .1, .3, .12 ]</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.10.7.2.1 [ .19 ]</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.35.1 [ .1, .4, .6 ]</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1 [ .2 ]</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>1.3.6.1.2.1.25.4.2.1 [ .1, .2, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ]</td>
<td></td>
</tr>
</tbody>
</table>
Linux

Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td>Privileges</td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td></td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td>hba_sysfs</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td>Privileges</td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ip_addr_ip</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>ip_link_mac</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td>Privileges</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ip_link_if</td>
<td>Privileges</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>rpmx</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsOf-i</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifOperStatus ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[.2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddAddr</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAdEntAddr, ipAdEntIfIndex, ipAdEntMediaType, ipAdEntNetMask ]</td>
<td>[.1, .2, .3]</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6AddrAddress, ipv6AddrPrefix ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifPhysAddress, ifOperStatus ]</td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalAddress, ipNetToPhysicalType ]</td>
<td>[.4, 6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.2, .4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort,</td>
<td>[.2, .3, .5, .6, .7, .8 ]</td>
</tr>
<tr>
<td></td>
<td>tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td>[.2, .3, .4]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry</td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td>[ udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpConnEntry</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnState, tcpConnLocalAddress, tcpConnLocalPort,</td>
<td>[.1, .2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>tcpConnRemAddress, tcpConnRemPort ]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td>IPV6-TCP-MIB::ipv6TcpConnEntry</td>
<td>[.1, .2, .3, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>[ ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress,</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td>ipv6TcpConnRemPort, ipv6TcpConnState ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[ udpLocalAddress, udpLocalPort ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IPV6-UDP-MIB::ipv6UdpConnEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6UdpLocalAddress, ipv6UdpLocalPort ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ]</td>
<td>[.1, .2, .3, .5, .6, .8 ]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.2.1.1.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifAlias, ifName, ifHighSpeed ]</td>
<td>[.1, .15, .18]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.2.1.26.2.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td>[.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.1.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td>[.19]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalAddress, ipNetToPhysicalType ]</td>
<td>[.1, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.1, .2, .4]</td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td>[.2]</td>
</tr>
</tbody>
</table>
### Method | MIB Values | OID
--- | --- | ---
getProcessList | HOST-RESOURCES-MIB::hrSWRunTable [hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters] | 1.3.6.1.2.1.25.4.2.1 [.1, .2, .4, .5]

### Mac OS X

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo'</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo'</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses'</td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifconfig_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>system_profiler-apps</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td>Privileges</td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo'</td>
<td>SNMPv2-MIB::sysDescr.0, SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.0, 1.3.6.1.2.1.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo'</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0, HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.1.1.0, 1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry [ifDescr, ifType, ifOperStatus], IP-MIB::ipAddressEntry [ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressMaskPrefix]</td>
<td>1.3.6.1.2.1.2.2.1 [.2, .3, .8], 1.3.6.1.2.1.4.31.2 [.2, .3, .4, .5]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
<td>-----</td>
</tr>
<tr>
<td>IP-MIB::ipAddrEntry</td>
<td>[ ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask ]</td>
<td>1.3.6.1.2.1.4.20.1 [.1, .2, .3]</td>
</tr>
<tr>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>[ ipv6AddrAddress, ipv6AddrPfxLength ]</td>
<td>1.3.6.1.2.1.55.1.8.1 [.1, .2]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.4.20.1 [.2, .3, .6, .8]</td>
</tr>
<tr>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>[ ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td>1.3.6.1.2.1.4.35.1 [.4 .6]</td>
</tr>
<tr>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>1.3.6.1.2.1.4.22.1 [.2, .4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1 [.2, .3, .5, .6, .7, .8]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.6.20.1 [.2, .3, .4]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.7.7.1 [.2, .3, .8]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1 [.1, .2, .3, .5, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.2.13.1 [.1, .2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.25.2.1 [.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.10.7.2.1 [.19]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.1.26.2.1.1 [.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.2.1.26.2.1.1 [.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.2.1.26.2.1.1 [.1, .3, .12]</td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1 [.2]</td>
</tr>
<tr>
<td>getProcessList</td>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>1.3.6.1.2.1.25.4.2.1 [.1, .2, .4, .5]</td>
</tr>
</tbody>
</table>
NetBSD

Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td></td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifconfig_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>pkg_info</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td></td>
</tr>
</tbody>
</table>

SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB:sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB:sysName.0</td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB:hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB:hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB:ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ifDescr, ifType, ifOperStatus ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB:ipAddressEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[.2, .3, .4, .5]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB:ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ifDescr, ifType, ifPhysAddress, ifOperStatus ]</td>
<td>[.1, .2, .3]</td>
</tr>
<tr>
<td></td>
<td>IPv6-MIB:ipv6AddrEntry</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>[ipv6AddrAddress, ipv6AddrPrefix ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB:ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.14.20.1</td>
</tr>
<tr>
<td></td>
<td>[ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB:ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.14.35.1</td>
</tr>
<tr>
<td></td>
<td>[ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td>[.4, .6]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemoteAddress, tcpConnectionRemotePort, tcpConnectionState, tcpConnectionProcess ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry</td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td>[ udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpConnEntry</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnState, tcpConnLocalAddress, tcpConnRemoteAddress, tcpConnRemotePort ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IPV6-TCP-MIB::ipv6TcpConnEntry</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6TcpConnLocalAddress, ipv6TcpConnRemoteAddress, ipv6TcpConnRemotePort ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td>[ udpLocalAddress, udpLocalPort ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IPv6-UDP-MIB::ipv6UdpEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6UdpLocalAddress, ipv6UdpLocalPort ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.2.3.1.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifAlias, ifName, ifHighSpeed ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.2.6.2.1.1</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.2.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalAddress, ipNetToPhysicalType ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToPhysicalAddress, ipNetToMediaType ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>1.3.6.1.2.1.25.4.2.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ]</td>
<td></td>
</tr>
</tbody>
</table>

OpenBSD

Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td>init</td>
</tr>
<tr>
<td>getDeviceInfo*</td>
<td>device_info</td>
<td></td>
</tr>
</tbody>
</table>
Method | Script | Privileges required
--- | --- | ---
getDirectoryListing | `ls` | Privileges
getFileContent | `file_content` | Privileges
getFileInfo | Handled by the `getFileMetadata` and `getFileContent` calls.
getFileMetadata | `file_metadata` | Privileges
getFileSystems | `df` | Privileges
getHostInfo | `host_info` | Privileges
getIPAddresses | `ifconfig_ip` | Privileges
getMACAddresses | `ifconfig_mac` | Privileges
getNetworkConnectionList | `netstat` | Privileges
getNetworkInterfaces | `ifconfig_if` | Privileges
getPackageList | `pkg_info` | Privileges
getProcessList | `ps` | Privileges
getProcessToConnectionMapping | `lsof-i` | Privileges

**SNMP**

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
</table>
| `getDeviceInfo` | SNMPv2-MIB:sysDescr.0
SNMPv2-MIB:sysName.0 | 1.3.6.1.2.1.1.1.0
1.3.6.1.2.1.1.5.0 |
| `getHostInfo` | HOST-RESOURCES-MIB:hrSystemUptime.0
HOST-RESOURCES-MIB:hrMemorySize.0 | 1.3.6.1.2.1.25.1.1.0
1.3.6.1.2.1.25.2.2.0 |
| `getIPAddresses` | IF-MIB::ifEntry
[ ifDescr, ifType, ifOperStatus ]
IP-MIB::ipAddressEntry
[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ] | 1.3.6.1.2.1.2.2.1
[ .2, .3, .8 ]
1.3.6.1.2.1.4.34.1
[ .2, .3, .4, .5 ] |
| `getMACAddresses` | IF-MIB::ifEntry
[ ifDescr, ifType, ifPhysAddress, ifOperStatus ]
IP-MIB::ipNetToPhysicalEntry
[ ipNetToPhysicalPhysAddress, ipNetToPhysicalType ] | 1.3.6.1.2.1.4.20.1
[ .1, .2, .3 ]
1.3.6.1.2.1.55.1.8.1
[ .1, .2 ] |
| `getNetworkConnectionList` | TCP-MIB::tcpConnectionEntry
[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemoteAddress, tcpConnectionRemotePort, tcpConnectionState, tcpConnectionProcess ]
TCP-MIB::tcpListenerEntry | 1.3.6.1.2.1.6.19.1
[ .2, .3, .5, .6, .7, .8 ] |
### Method Values

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP-MIB::tcpConnEntry</td>
<td>[ tcpConnState, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort ]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td>IPv6-TCP-MIB::ipv6TcpConnEntry</td>
<td>[ ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState ]</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td>UDP-MIB::udpConnEntry</td>
<td>[ udpLocalAddress, udpLocalPort ]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td>IPv6-UDP-MIB::ipv6UdpEntry</td>
<td>[ ipv6UdpLocalAddress, ipv6UdpLocalPort ]</td>
<td>1.3.6.1.2.1.6.17.5</td>
</tr>
</tbody>
</table>

### OpenVMS

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>deviceinfo</td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>show-mount</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>hostinfo</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip, tcpip_show_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>tcpip_show_mac</td>
<td></td>
</tr>
</tbody>
</table>
### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysDescr.0, SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0, HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry[ifDescr, ifType, ifOperStatus], IP-MIB::ipAddrEntry[ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 8]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 4, 5]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry[ifDescr, ifType, ifPhysAddress, ifOperStatus], IP-MIB::ipNetToPhysicalAddress[ipNetToPhysicalPhysAddress, ipNetToPhysicalType]</td>
<td>1.3.6.1.2.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 6, 8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry[ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>1.3.6.1.2.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[4, 6]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry[tpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess], TCP-MIB::tcpListenerEntry[tpListenerLocalAddress, tpListenerLocalPort, tpListenerProcess], UDP-MIB::udpEndpointEntry[udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess]</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 5, 6, 7, 8 ]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 4]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 3, 8]</td>
</tr>
<tr>
<td></td>
<td>IPv6-TCP-MIB::ipv6TcpConnEntry[ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2, 3, 4, 5]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2, 3, 4, 6]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1, 2]</td>
</tr>
</tbody>
</table>
### Method | MIB Values | OID
---|---|---
getNetworkInterfaces | IF-MIB::ifEntry
[ ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ] | 1.3.6.1.2.1.2.2.1
| IF-MIB::ifXEntry
[ ifAlias, ifName, ifHighSpeed ] | 1.3.6.1.2.1.31.1.1.1
| MAU-MIB::ifMauEntry
[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ] | 1.3.6.1.2.1.26.2.1.1
| EtherLike-MIB::dot3StatsEntry
[ dot3StatsDuplexStatus ] | 1.3.6.1.2.1.10.7.2.1
| IP-MIB::ipNetToPhysicalEntry
[ ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress, ipNetToPhysicalType ] | 1.3.6.1.2.1.4.35.1
| IP-MIB::ipNetToMediaEntry
[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ] | 1.3.6.1.2.1.4.22.1

getPackageList | HOST-RESOURCES-MIB::hrSWInstalledTable
[ hrSWInstalledName ] | 1.3.6.1.2.1.25.6.3.1

getProcessList | HOST-RESOURCES-MIB::hrSWRunTable
[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ] | 1.3.6.1.2.1.25.4.2.1

### POWER HMC

#### Shell scripts

| Method | Script | Privileges required |
---|---|---|
initialise | init | |
getDeviceInfo | device_info | *
getFileSystems | monhmc-rdisk | |
getHostInfo | host_info | |
getIPAddresses | lshmc-n | *
getMACAddresses | netstat-ine | |
getNetworkConnectionList | netstat | |
getNetworkInterfaces | lshmc_netstat | |

### SNMP

| Method | MIB Values | OID |
---|---|---|
getDeviceInfo | SNMPv2-MIB::sysDescr.0
SNMPv2-MIB::sysName.0 | 1.3.6.1.2.1.1.1.0
1.3.6.1.2.1.1.5.0 |
getHostInfo | HOST-RESOURCES-MIB::hrSystemUptime.0
HOST-RESOURCES-MIB::hrMemorySize.0 | 1.3.6.1.2.1.25.1.1.0
1.3.6.1.2.1.25.2.2.0 |
getIPAddresses | |

---

BMC Atrium Discovery 10.2  Page 524 of 1671
<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry[ifDescr, ifType, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry[ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix]</td>
<td>1.3.6.1.2.1.4.341</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrEntry[ipAddrAddr, ipAddrIfIndex, ipAddressType, ipAddressPrefix]</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>IPv6-MIB::ipv6AddrEntry[ipv6AddrAddress, ipv6AddrPrefix]</td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.1</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry[ifConnectionLocalAddress, ifConnectionLocalPort, ifConnectionRemAddress, ifConnectionRemPort, ifConnectionState, ifConnectionProcess]</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry[ifListenerLocalAddress, ifListenerLocalPort, ifListenerProcess]</td>
<td>1.3.6.1.2.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry[udpLocalAddress, udpLocalPort]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>_IPV6-UDP-MIB::ipv6UdpEntry[ipv6UdpLocalAddress, ipv6UdpLocalPort]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.1</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry[ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry[ifAlias, ifName, ifHighSpeed]</td>
<td>1.3.6.1.2.1.2.1.2.1.1</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry[ifMauIndex, ifMauType, ifMauAutoNegSupported]</td>
<td>1.3.6.1.2.1.2.1.2.1.1.1</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry[dot3StatsDuplexStatus]</td>
<td>1.3.6.1.2.1.2.1.2.1.2.1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.1</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry[ifNetToPhysicalPhysAddress, ifNetToPhysicalType]</td>
<td>1.3.6.1.2.1.2.1.2.1.2.1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.1</td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable[hrSWInstalledName]</td>
<td>1.3.6.1.2.1.25.6.3.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.2</td>
</tr>
</tbody>
</table>
### Solaris

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td>Handled by the getMetadata and getContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td>hba_fcinfo</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>hba_emlxadm</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>hba_hbacmd</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>hba_lputil</td>
<td>Privileges</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td>Privileges</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>netstat_mac</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>interface_commands</td>
<td>Privileges</td>
</tr>
<tr>
<td>getPackageList</td>
<td>pkginfo</td>
<td></td>
</tr>
<tr>
<td>getPatchList</td>
<td>patch_list</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td>Privileges</td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsofi-i</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>pfiles</td>
<td>Privileges</td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ifDescr, ifType, ifOperStatus ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[.2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td>[ipv6AddrAddress, ipv6AddrPfxLength ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ifDescr, ifType, ifPhysAddress, ifOperStatus ]</td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ipNetToPhysicalPhysAddress, ipNetToPhysicalType]</td>
<td>[.4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>[.2, .4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[tcpConnectionLocalAddress, tcpConnectionLocalPort,</td>
<td>[.2, .3, .5, .6, .7, .8]</td>
</tr>
<tr>
<td></td>
<td>tcpConnectionRemAddress, tcpConnectionRemPort,</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>tcpConnectionState, tcpConnectionProcess ]</td>
<td>[.2, .3, .4]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.7.7.1</td>
</tr>
<tr>
<td></td>
<td>[tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpEndpointEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpConnEntry</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>[tcpConnState, tcpConnLocalAddress, tcpConnLocalPort,</td>
<td>[.1, .2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>tcpConnRemAddress, tcpConnRemPort ]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td>IPV6-TCP-MIB::ipv6TcpConnEntry</td>
<td>[.1, .2, .3, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>[ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort,</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td>ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[udpLocalAddress, udpLocalPort]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IPV6-UDP-MIB::ipv6UdpEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ipv6UdpLocalAddress, ipv6UdpLocalPort]</td>
<td>[.1, .2, .3, .5, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.31.1.1.1</td>
</tr>
<tr>
<td></td>
<td>[ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ]</td>
<td>[.1, .15, .18]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.26.2.1.1</td>
</tr>
<tr>
<td></td>
<td>[ifAlias, ifName, ifHighSpeed]</td>
<td>[.1, .3, .12]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td>[ifMauIfIndex, ifMauType, ifMauAutoNegSupported]</td>
<td>[.19]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[dot3StatsDuplexStatus ]</td>
<td>[.1, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress,</td>
<td>[.1, .2, .4]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>ipNetToPhysicalType ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[  ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType</td>
<td></td>
</tr>
<tr>
<td>getPackages</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1 [ .2 ]</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>1.3.6.1.2.1.25.4.2.1 [ .1, .2, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUN-SNMP-MIB::psEntry</td>
<td>1.3.6.1.4.1.42.3.12.1 [ .1, .2, .8, .9, .10 ]</td>
</tr>
<tr>
<td></td>
<td>[ psProcessID, psParentProcessID, psProcessUserName,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>psProcessUserID, psProcessName ]</td>
<td></td>
</tr>
</tbody>
</table>

### Tru64

#### Shell scripts

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo*</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileInfo</td>
<td>file_content</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Handled by the getFileInfoMetadata and getFileInfoContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo*</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses*</td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifconfig_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>setld</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td></td>
</tr>
</tbody>
</table>

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifOperStatus ]</td>
<td>[ .2, .3, .8 ]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrAddressEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[ .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td></td>
<td>[ ipv6AddrAddress, ipv6AddrPfxLength ]</td>
<td>[ .1, .2 ]</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifPhysAddress, ifOperStatus ]</td>
<td>[ .2, .3, .6, .8 ]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td>[ .4, .6 ]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.2.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess ]</td>
<td>[ .2, .3, .5, .6, .7, .8 ]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.2.6.20.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td>[ .2, .3, .4 ]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.2.6.7.1</td>
</tr>
<tr>
<td></td>
<td>[ udpLocalAddress, udpLocalPort ]</td>
<td>[ .2, .3, .8 ]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus ]</td>
<td>[ .1, .2, .3, .5, .6, .8 ]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifAlias, ifName, ifHighSpeed ]</td>
<td>[ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td>[ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td>[ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress, ipNetToPhysicalType ]</td>
<td>[ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.2.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td>[.2]</td>
</tr>
<tr>
<td>getProcessList</td>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>1.3.6.1.2.1.25.4.2.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ]</td>
<td>[.1, .2, .4, .5]</td>
</tr>
</tbody>
</table>

**UnixWare**

**Shell scripts**

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>ifconfig_ip</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>ifconfig_mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>netstat</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>ifconfig_if</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>pkginfo</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td>Privileges</td>
</tr>
</tbody>
</table>

**SNMP**

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, ifOperStatus ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry</td>
<td>1.3.6.1.2.1.4.34.1</td>
</tr>
<tr>
<td></td>
<td>[ ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix ]</td>
<td>[.2, .3, .4, .5]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>IP-MIB::ipAddrEntry</td>
<td>[ ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask ]</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[.1, .2, .3]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.3.6.1.2.1.55.1.8.1</td>
</tr>
<tr>
<td>IPV6-MIB::ipv6AddrEntry</td>
<td>[ ipv6AddrAddress, ipv6AddrPfxLength ]</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td>getMACAddresses *</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.4.20.1</td>
</tr>
<tr>
<td></td>
<td>[ ifDescr, ifType, iPhysAddress, ifOperStatus ]</td>
<td>[.2, .3, .6, .8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalPhysAddress, ipNetToPhysicalType]</td>
<td>[.4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.2, .4]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnectionEntry</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess ]</td>
<td>[.2, .3, .5, .6, .7, .8]</td>
</tr>
<tr>
<td></td>
<td>TCP-MIB::tcpListenerEntry</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td></td>
<td>[ tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess ]</td>
<td>[.2, .3, .4]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnEntry</td>
<td>1.3.6.1.2.1.7.1.7.1</td>
</tr>
<tr>
<td></td>
<td>[ udpLocalAddress, udpLocalPort, udpLocalProcess ]</td>
<td>[.2, .3, .8]</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>IF-MIB::ifEntry</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td></td>
<td>[ ifIndex, ifDescr, ifType, ifSpeed, iPhysAddress, ifOperStatus ]</td>
<td>[.1, .2, .3, .4, .5]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifXEtry</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td></td>
<td>[ ifAlias, ifName, ifHighSpeed ]</td>
<td>[.1, .2, .3, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>MAU-MIB::ifMauEntry</td>
<td>1.3.6.1.2.1.7.5.1</td>
</tr>
<tr>
<td></td>
<td>[ ifMauIfIndex, ifMauType, ifMauAutoNegSupported ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td></td>
<td>[ dot3StatsDuplexStatus ]</td>
<td>[.1, .2]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>1.3.6.1.2.1.10.7.2.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToPhysicalIfIndex, ipNetToPhysicalAddress, ipNetToPhysicalType ]</td>
<td>[.19]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.35.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.1, .4, .6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>1.3.6.1.2.1.4.22.1</td>
</tr>
<tr>
<td></td>
<td>[ ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType ]</td>
<td>[.1, .2, .4]</td>
</tr>
<tr>
<td>getPackageList</td>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>1.3.6.1.2.1.25.6.3.1</td>
</tr>
<tr>
<td></td>
<td>[ hrSWInstalledName ]</td>
<td>[.2]</td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td>1.3.6.1.2.1.25.4.2.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[.1, .2, .4, .5]</td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
|                 | HOST-RESOURCES-MIB::hrSWRunTable  
[ hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters ] |                                                                      |

VMware ESX

vSphere

<table>
<thead>
<tr>
<th>Method</th>
<th>Managed Object Reference</th>
<th>Properties</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>HostSystem</td>
<td>name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ServiceInstance</td>
<td>content.about</td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>Datastore</td>
<td>summary.name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.capacity</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.freeSpace</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.url</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.vmfs.uuid</td>
<td>VMFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.remoteHost</td>
<td>NFS &amp; CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.remotePath</td>
<td>NFS &amp; CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.userName</td>
<td>CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.path</td>
<td>LOCAL volumes only</td>
</tr>
<tr>
<td>getHBAInfo</td>
<td>HostSystem</td>
<td>config.storageDevice.hostBusAdapter[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].nodeWorldWideName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].portWorldWideName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].model</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].driver</td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td>HostSystem</td>
<td>hardware.memorySize</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.model</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.vendor</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.uuid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.otherIdentifyingInfo[].identifierType.key</td>
<td>Not used if unset</td>
</tr>
</tbody>
</table>

BMC Atrium Discovery 10.2
<table>
<thead>
<tr>
<th>Method</th>
<th>Managed Object Reference</th>
<th>Properties</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.otherIdentifyingInfo[].identifierValue</td>
<td>Not used if unset or &quot;unknown&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuPkg.threadId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuPkg.numCpuCores</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuInfo.threadId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.hyperThread.active</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.dnsConfig.dnsDomainName</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.dnsConfig.hostName</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>HostSystem</td>
<td>config.network.pnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ipV6Config.ipV6Address[].ipAddress</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ipV6Config.ipV6Address[].ipAddress</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ipV6Config.ipV6Address[].ipAddress</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td>getPatchList</td>
<td>HostSystem</td>
<td>configManager.patchManager</td>
<td>Uses queryHostPatch() task</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>HostSystem</td>
<td>config.network.pnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].driver</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.linkSpeed.speedMb</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Managed Object Reference</td>
<td>Properties</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.linkSpeed.duplex</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses *</td>
<td>HostSystem</td>
<td>config.network.pnic[].mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td>getVirtualMachines</td>
<td>VirtualMachine</td>
<td>config.name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.uuid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.guestFullName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.guestId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.alternateGuestName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>name</td>
<td>Used if config.name is empty</td>
</tr>
<tr>
<td></td>
<td></td>
<td>guest.guestId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>guest.guestFullName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>runtime.powerState</td>
<td></td>
</tr>
</tbody>
</table>

*indicates methods that must succeed for a Host to be created.

**Shell scripts**

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHBAList</td>
<td>hba_sysfs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hba_procfs</td>
<td>Privileges</td>
</tr>
<tr>
<td></td>
<td>hba_hbacmd</td>
<td></td>
</tr>
</tbody>
</table>
### Method

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>hba_lputil</td>
<td></td>
<td></td>
</tr>
<tr>
<td>host_info</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ip_addr_ip</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ifconfig_ip</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ip_link_mac</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ifconfig_mac</td>
<td></td>
<td></td>
</tr>
<tr>
<td>netstat</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ip_link_if</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ifconfig_if</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rpm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rpmx</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dpkg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getHostInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getProcessToConnectionMapping</td>
<td>lsof-i</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Privileges</td>
</tr>
</tbody>
</table>

### VMware ESXi

### vSphere

<table>
<thead>
<tr>
<th>Method</th>
<th>Managed Object Reference</th>
<th>Properties</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>HostSystem</td>
<td>name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ServiceInstance</td>
<td>content.about</td>
<td></td>
</tr>
<tr>
<td>getFileSystems</td>
<td>Datastore</td>
<td>summary.name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.type</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.capacity</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.freeSpace</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>summary.url</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.vmfs.uuid</td>
<td>VMFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.remoteHost</td>
<td>NFS &amp; CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.remotePath</td>
<td>NFS &amp; CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.nas.userName</td>
<td>CIFS volumes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>info.path</td>
<td>LOCAL volumes only</td>
</tr>
<tr>
<td>getHBAInfo</td>
<td>HostSystem</td>
<td>config.storageDevice.hostBusAdapter[].device</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Managed Object Reference</td>
<td>Properties</td>
<td>Notes</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].nodeWorldWideName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].portWorldWideName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].model</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.storageDevice.hostBusAdapter[].driver</td>
<td></td>
</tr>
<tr>
<td>getHostInfo²</td>
<td>HostSystem</td>
<td>hardware.memorySize</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.model</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.vendor</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.uuid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.otherIdentifyingInfo[].identifierType.key</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.systemInfo.otherIdentifyingInfo[].identifierValue</td>
<td>Not used if unset or &quot;unknown&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuPkg.threadId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuPkg.numCpuCores</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>hardware.cpuInfo.threadId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.hyperThread.active</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.dnsConfig.dnsDomainName</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.dnsConfig.hostName</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>HostSystem</td>
<td>config.network.pnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.ipV6Config.ipV6Address[].ipAddress</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.ip.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.ipV6Config.ipV6Address[].ipAddress</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.ip.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].device</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Managed Object Reference</td>
<td>Properties</td>
<td>Notes</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ip.ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ip.subnetMask</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ipV6Config.ipV6Address[].ipAddress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.ipV6Config.ipV6Address[].prefixLength</td>
<td>Not used if unset</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>HostSystem</td>
<td>config.network.pnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].driver</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.linkSpeed.speedMb</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.pnic[].spec.linkSpeed.duplex</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].device</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>HostSystem</td>
<td>config.network.pnic[].mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.vnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.network.consoleVnic[].spec.mac</td>
<td></td>
</tr>
<tr>
<td>getPatchList</td>
<td>HostSystem</td>
<td>configManager.patchManager</td>
<td>Uses queryHostPatch() task</td>
</tr>
<tr>
<td>getVirtualMachines</td>
<td>VirtualMachine</td>
<td>config.name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.uuid</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.guestFullName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.guestId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>config.alternateGuestName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>name</td>
<td>Used if config.name is empty</td>
</tr>
<tr>
<td></td>
<td></td>
<td>guest.guestId</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>guest.guestFullName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>runtime.powerState</td>
<td></td>
</tr>
</tbody>
</table>

**Shell scripts**
<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Privileges required</th>
</tr>
</thead>
<tbody>
<tr>
<td>initialise</td>
<td>init</td>
<td></td>
</tr>
<tr>
<td>getDeviceInfo`</td>
<td>device_info</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>ls</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td>file_content</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Handled by the getFileMetadata and getFileContent calls.</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td>file_metadata</td>
<td>Privileges</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>df</td>
<td></td>
</tr>
<tr>
<td>getHostInfo`</td>
<td>host_info</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>vim-cmd-addresses</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses`</td>
<td>vim-cmd-mac</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>vim-cmd-if-details</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>ps</td>
<td></td>
</tr>
</tbody>
</table>

### Standalone UNIX scanning

A scanner file is a plain text file that is used to simulate the discovery of a system that is unreachable, or one that you are not permitted to scan. You create a scanner file by running the standard discovery commands on a host and saving the output. Only the standard discovery commands are run on the host; information that is discovered by patterns is not available.

### Scanning Windows targets

For Windows targets, you cannot download discovery commands. To discover Windows targets you must use the Standalone Windows scanning tool.

### Scanner files and clusters

Clusters do not support scanner files directly. To get information from a scanner file into a cluster, you must use the scanner file with a standalone machine which consolidates to a cluster.

### Process for using scanner files

The process for using scanner files is:

1. Create a scanner file.
   a. Download the "*.sh file from the BMC Atrium Discovery user interface."
1. **b.** Copy the file to the remote host.
   **c.** Run and capture the output to a file.

2. **Load the scanner file onto the appliance.**
   **a.** Using the SCP utility, upload the results file to the appliance as the upload user.

### Creating a scanner file

**To create a scanner file:**

1. From the Discovery section of the **Administration** tab, select **Platforms**.
2. Click the OS link corresponding to the host for which you are creating the scanning file.
   The commands for the OS are displayed. The following example illustrates creating a scanner file using the Linux OS.

   ![Diagram illustrating a scanning file created for a Linux OS.](image)

   **This diagram illustrates a scanning file created for a Linux OS.**

3. **Click Download host script** (named Linux.sh, in this example) and save the file to the local computer as an executable.
4. **Copy the file to the remote host.**
   In the following example, the SCP utility is used to copy the files between the local host `teaspoon` and the remote host `teabag`:

   ```bash
   tideway@teaspoon:~$ scp linux.sh tideway@teabag:linux.sh
   tideway@teabag's password:
   linux.sh                                      100%   19KB  18.9KB/s   00:00
   tideway@teaspoon:~$
   ```

5. **Log on to the remote host and run the script, piping the output into a text file:**

   ```bash
   tideway@teaspoon:~$ ssh teabag
   tideway@teabag's password:
   Linux teabag 2.6.18-6-686 #1 SMP Sat Dec 27 09:31:05 UTC 2008 i686
   ...
   tideway@teabag:~$ ./linux.sh > teabag.txt
   tideway@teabag:~$ more teabag.txt
   FORMAT Linux
   --- START device_info
   ```
hostname: teabag
fqdn: teabag
dns_domain: tideway.com
domain:
os: Debian Linux lenny/sid
--- END device_info
--- START host_info
kernel: 2.6.25-2-amd64
num_logical_processors: 2
cores_per_processor: 2
...
tideway@teabag:~$

6. Copy the output text file to your local host.

tideway@teabag:~$
  tideway@teabag:~$ scp teabag.txt tideway@teaspoon:linux.txt
tideway@teaspoon's password: teabag.txt 100% 265KB 264.7KB/s 00:00
tideway@teabag:~$

The output file is copied to your local host.

Loading the scanner file onto the appliance

After you have created and copied the output file to your local host, you can copy it to the appliance. In this example, the appliance is called appliance, and the local host is called teaspoon. When loading scanner files onto the appliance, use the upload user, because it has a home directory that is the default location (/usr/tideway/var/scanner) for uploading scanner files, and it has limited access to the appliance. See The upload user for information on enabling this user.

1. Ensure the file has the group read permission set:

dtweed@teaspoon:~$ chmod 640 teabag.txt
dtweed@teaspoon:~$

2. Copy the output file to the appliance. In this example, the SCP utility is used:

dtweed@teaspoon:~$ scp -p teabag.txt upload@appliance:~/linux.teabag.txt
upload@appliance's password: teabag.txt 100% 265KB 262.7KB/s 00:00
dtweed@teaspoon:~$
When you load a scanner file onto the appliance, its name must be unique; otherwise, it might get overwritten by another scanner file being uploaded at the same time. For this reason, it is helpful to use a naming scheme that enables you to correlate scanner files and created hosts. Do not use a name starting with . or ending with .ignore. If you do, that file will be ignored. File names are used only for internal purposes.

⚠️ Using scanner files with consolidating appliances

If you are using scanner files with consolidating appliances, upload the scanner files to the consolidating appliance, rather than to the scanning appliance. Doing so correctly identifies the hosts as "Read from scanner file", rather than as "Retrieved by scanning appliance".

After a scanner file is loaded, you can look at the results of the discovery and view the host, as shown in the following illustration.

This diagram illustrates the properties displayed for a scanned host.

On the DiscoveryAccess page, in the Data Source field of the Discovery Details section "Read from scanner file" is displayed. This is shown in the following illustration.

This diagram illustrates Read from scanner file displaying on the DiscoveryAccess page.
Considerations when using scanner files

BMC Atrium Discovery initiates a scan of the IP address automatically. You do not need to initiate a scan manually. However, depending on how you set the Scanner File Polling Interval option, it can take up to an hour after the file is copied to the appliance for the scan to initiate. Scanner files are read from the `$TIDEWAY/var/scanner` directory periodically based on this setting. For more information on setting this option, see Scanning settings.

When scanner files are used, BMC Atrium Discovery creates non-expiring pool data by creating a `.no-expiry` file in the pool data directory `$TIDEWAY/var/pool/192/168/1/100/.no-expiry`, where the IP address is `192.168.1.100`. The pool data **will always be used** for each subsequent discovery and return identical results until the scanner file is updated, or until the pool data is deleted.

- Hosts discovered by scanner files never age.
- BMC Atrium Discovery does not handle overlapping IP ranges, which occur in rare cases. Using the Windows scanning tool might increase the likelihood of ranges overlapping; however, the tool enables you to select a specific IP to avoid one you have already selected. For more information, see Standalone Windows scanning tool.

Discovering Windows Hosts

To discover Windows hosts, BMC Atrium Discovery uses one or more *Windows proxies*. Once the BMC Atrium Discovery appliance decides that a discovery target is running a Windows operating system, it uses a proxy to interrogate the target. Often, the proxy is also responsible for providing authentication and authorization of the discovery activities.

There are two types of proxy:

- **Active Directory Proxy** — runs as an Active Directory user, and uses those user credentials to connect to Windows hosts within the Active Directory domain. Credentials are not stored in the BMC Atrium Discovery credential vault.

- **Credential Proxy** — runs as a local administrator user. Credentials are stored in the BMC Atrium Discovery credential vault and are provided to the proxy as required.

A single Windows host may run both types of proxy. To handle complex Active Directory environments, it is possible to run multiple Active Directory proxies as different users. The Active Directory proxy can also be used in a legacy Windows Workgroup environment to connect to workgroup members using the proxy's workgroup credentials.

The *Proxy manager* is used to manage the running proxies and their configuration, and to establish secure connections with approved BMC Atrium Discovery appliances.
Installing and running Windows proxies

Installing or upgrading Windows proxies where anti-virus software is installed

Before installing Windows discovery proxies you should either disable the anti-virus software or configure it to exclude RemQuery from triggering a virus alert. You can enable the anti-virus software once the Windows proxy has been installed.

The Windows Proxy installer is downloaded from the BMC Atrium Discovery appliance user interface. Visit Discovery > Tools and download the installer. Both kinds of proxy, and the proxy manager, are in the same installer.

The installer prompts for the installation directory and whether to create start menu items. It also permits you to choose whether to create Active Directory and Credential proxies. In each case, credentials for a suitable user may be provided. If you do not create the proxies, or you do not enter credentials at this stage, you may do so later using the Proxy Manager. If proxies are created at this stage, the installer gives the option of registering the proxies with the appliance from which the installer was downloaded. Registration in this way opens a web browser displaying the appliance UI, and is therefore only possible if the Windows proxy host has web access to the appliance.

To modify the proxies that are running, or add new ones, run the Proxy Manager from the Start menu, by default under BMC Software > ADDM Proxy > Proxy Manager.

The Active Directory proxy must be given the credentials of an Active Directory user. Ideally the user should be a domain administrator. If that is not possible, the user must have a range of permissions, and discovery capabilities will be limited. The Credential proxy should run as a local Windows user with administrator privileges.

Each proxy listens on a particular port. The default ports are 4321 for the Active Directory proxy and 4323 for the Credential proxy. Whichever ports you choose must be reachable from the BMC Atrium Discovery appliance, so the Windows firewall and any intermediate network firewalls must allow connections from the appliance’s IP address.

When scanning a Windows discovery target, the BMC Atrium Discovery appliance chooses a Proxy Pool based on the target’s IP address. Often, a pool contains a single proxy, but if the proxy becomes a bottleneck, a pool can contain several proxies.

Connecting a Windows proxy to an appliance

For security, the proxy and the appliance must exchange certificates and each end must confirm the connection.
If it is possible to access the appliance web user interface from the Windows machine running the proxy, you may set up the connection between appliance and proxy from the Proxy Manager user interface:

1. In the Proxy Manager, choose Edit menu > Known Appliances
2. Press the green plus icon to add an appliance
3. Enter the appliance address, and select Contact
4. The proxy manager exchanges certificates with the appliance and displays the appliance fingerprint. Assuming the fingerprint is correct, select Register
5. A browser window opens, displaying the Create Proxy page. Confirm the details and select Apply

If there is no web access from the Windows proxy machine to the appliance, the proxy connection must be initiated from the appliance user interface and confirmed in the Proxy Manager:

1. In the appliance web user interface, visit Discovery > Credentials > Windows Proxies
2. If need be, add a new Proxy Pool using the Add... button
3. In the chosen Proxy Pool, select Actions > Add Windows Proxy
4. Enter a proxy name and address, and select Apply. The user interface shows Approve this appliance in the proxy manager
5. In the Proxy Manager, choose Edit menu > Known Appliances
6. The new appliance is shown in orange with a status Pending approval
7. Double click the entry then choose Approve

Configuring Windows discovery

BMC Atrium Discovery is a Linux-based appliance. Because the methods that are used to access Windows hosts are only available from Windows systems, Windows discovery requires a Windows proxy host.

Windows proxy

The Windows proxy scans Windows hosts on behalf of the discovery service on the BMC Atrium Discovery appliance.

Proxy changes in 8.3 SP2 and later versions

In BMC Atrium Discovery 8.3 SP2 and later versions, installation and management of proxies has been improved with the introduction of a new Windows proxy manager tool. It is also possible to install multiple proxies of each type on a single host. For information about the changes, see Windows proxy manager.
You can download the Windows proxies and Windows proxy manager as installation files from the appliance and install onto the local Windows host. For more information, see Installing Windows proxies.

Windows discovery is handled by one of the following ways:

- **Credential Windows proxy** — It is a BMC Atrium Discovery service that runs on a customer-provided Windows host. To perform discovery, it uses credentials supplied by the BMC Atrium Discovery appliance from the credentials vault.
- **Active Directory Windows proxy** — It is a BMC Atrium Discovery service that runs on a customer-provided Windows host. To perform discovery, it logs in as an Active Directory user.
  - While installing the proxy, you must configure it as a user on the Active Directory domain that can log in and run discovery commands on the hosts to discover. The Active Directory proxies do not use the credentials supplied by the BMC Atrium Discovery appliance from the credentials vault.
  - When you install the Active Directory Windows proxy (as the Windows domain administrator), the appliance uses it to discover the Windows hosts in that domain. The proxy can only discover Windows hosts on the domain it is a member of, or other domains trusted by that domain. To discover domains which are not trusted, you must configure another Windows proxy with the appropriate domain permissions.

From BMC Atrium Discovery version 8.2, the Workgroup Windows proxy is no longer supplied with the appliance. All of its functionality has been moved into the Active Directory Windows proxy.

**Windows proxy manager**

The Windows proxy manager enables you to install and manage proxies on the Windows host on which the manager is installed. The Windows proxy manager is installed when you install a proxy. You can perform the following tasks using the Windows proxy manager:

- Create (install a new proxy service)
- Edit the port that the proxy uses and the user account that the proxy runs as
- Delete (uninstall a proxy service)
- Start a selected proxy
- Stop a selected proxy
- Restart a selected proxy

**Windows proxy pool**

To balance the load of the proxies, distribute discovery requests, and offer scalability and better performance solutions for Windows discovery, proxies are grouped into proxy pools based on the following criteria:
• Type of proxies: A proxy pool must contain either Credential proxies or Active Directory proxies. A proxy pool must not contain proxies of both the types.

• Version of proxies: A proxy pool must contain either version 9.0 proxies or proxies of earlier versions. A proxy pool must not contain proxies of both version 9.0 and proxies of earlier versions.

Based on the proxy version and version of the Operating System (OS) the proxy runs on, the proxy capability is one of the following:

• Fully IPv6 capable: Can scan IPv6 addresses and retrieve IPv6 data. Where BMC Atrium Discovery version 9.0 or later proxies are running on Windows 2008 or later.

• Cannot scan IPv6 addresses: Can retrieve IPv6 data but the Windows version does not support scanning IPv6 addresses. Where BMC Atrium Discovery version 9.0 or later proxies are running on versions of Windows older than Windows 2008.

• Not IPv6 capable: Cannot scan IPv6 addresses and cannot retrieve IPv6 data. BMC Atrium Discovery proxies from versions older than 9.0.

The proxies in a pool must have identical access to Windows hosts, as only one proxy per pool is tested for access. The appliance’s UI displays the pools in the order (from top to bottom) in which you have added them to the appliance. You can change their order. For discovery tasks, the proxies in a Windows proxy pool are selected depending on their loading. If a proxy is overloaded, or unavailable, the discovery task is assigned to the next available proxy in the pool.

Operating System compatibility for IPv6 discovery

To discover IPv6 hosts, the Operating System (OS) and proxy compatibility requirements are as follows:

• The version of the proxy must be of BMC Atrium Discovery 9.0, or later version

• The proxy must run on an OS which is Windows 2008, or later version

• The OS of the target hosts must be Windows 2008, Windows Vista, or a later version

The supported discovery methods are WMI and RemQuery.

Steps to configure Windows discovery

Before you can use a Windows proxy to discover your Windows IT infrastructure, you must perform the following tasks in the given order:

1. Download and install the Windows proxy software (including the Windows proxy manager). For more information, see installing Windows proxies.

2. Add Windows proxies to the appliance using the Windows proxy manager. For more information, see Windows proxy manager.

3. Add Windows proxy pools. For more information, see adding proxy pools.

4. Add Windows proxies to the proxy pools. For more information, see adding proxies to pools.
5. Edit the firewall rules to permit communication between the appliance and the Windows proxies.
   For more information, see editing the firewall.
6. Verify that the Windows proxy service has started.
   For more information, see testing Windows credentials and communication.

⚠️ Potential user lock out

By default, AD accounts have a limited number of login attempts, for example three attempts in fifteen minutes. Access Denied errors from WMI, DCOM, and local commands such as systeminfo are counted as unsuccessful login attempts. Where target hosts are incorrectly configured, this limit can be exceeded and the account locked out.

To avoid this, configure the Discovery account to accept unlimited login attempts.

Windows discovery utilities no longer shipped in BMC Atrium Discovery

From version 8.1 the following utilities are no longer shipped with BMC Atrium Discovery. You can download them freely and install them where required.

- **rcmd**: see the Microsoft Download Center and search for the resource kit for the version of Windows that you are running.
- **pulist**: download this from the Microsoft Download Center.
- **tlist**: download this from the Microsoft Download Center.

The Windows proxy configuration file

The Windows proxy is configured using a configuration file (*winproxy.conf*), which is located at:

```
C:\Program Files\BMC Software\ADDM\Proxy\runtime\<Proxy_Name>\etc\winproxy.conf
```

where `<Proxy_Name>` is the type of the Windows proxy (AD Proxy or Credential).

You can download *winproxy.conf* files from Windows proxies that are connected to your appliance. Any modifications can be made to the file which can then be uploaded to the Windows proxy or to multiple Windows proxies. This simplifies the management of your Windows proxies. See Windows proxy configuration for more information.

- Access methods
- Optional commands
- Commands to run
- Checksum
- Example Windows proxy configuration file part
Access methods

This section is a list of access methods, in the order in which they should be attempted. In this section you can enable or disable an access method. For example:

```xml
<setting name="wmi" enabled="True"/>
```

or

```xml
<setting name="wmi" enabled="False"/>
```

The access methods available are:

- `wmi` - WMI commands.
- `remquery` - RemQuery commands.

Optional commands

This section lists the additional executables which are available on the Windows proxy. These are generally tools which are not standard on all versions of Windows, or are third-party supplied tools such as the PSINFO utility described in User privileges and information access for Windows operating systems. In this section you can enable or disable the optional commands. For example:

```xml
<setting name="tcpvcon" enabled="True"/>
```

or

```xml
<setting name="tcpvcon" enabled="False"/>
```

Commands to run

This section lists the commands and provides the actual command string that is executed. The commands are referred to by the setting name defined in the previous section. For example:

```xml
<command name="tcpvcon">tcpvcon -anc</command>
```

or
Checksum

In this section a checksum is written to ensure that the file has not been tampered with since being copied or uploaded from the appliance.

```
...<checksum>6d263f1f630e934a3180fb182bc589a8</checksum>
</slave>
```

You can add the checksum without uploading the file using the `tw_sign_winproxy_config` utility. You can then copy the signed file to multiple appliances using ftp or similar.

Example Windows proxy configuration file part

```
<slave>

 <!-- These settings control the various access methods -->
<setting name="wmi" enabled="True"/>
<setting name="remquery" enabled="True"/>

 <!-- These settings control the use of optional commands -->
<setting name="wmi-query-creation" enabled="True"/>
<setting name="openports" enabled="False"/>
<setting name="tcpvcon" enabled="True"/>

 <!-- These are the commands that will be run -->
<command name="systeminfo">echo begin wmic_values: &amp;&amp; wmic bios get serialnumber 2>&amp;1 &amp; wmic csproduct get uuid 2>&amp;1 &amp; echo end wmic_values: &amp;&amp; systeminfo /fo csv /nh</command>
<command name="infocmds">hostname &amp;&amp; hostname &amp;&amp; ver</command>
<command name="ipconfig">ipconfig</command>
<command name="tasklist">tasklist /fo csv /nh /v</command>
<command name="netstat">netstat</command>
<command name="openports">openports.exe</command>
<command name="tcpvcon">tcpvcon.exe -anc</command>

<!-- lputil commands for Emulex HBAs -->
<command name="lputil-listhbas">lputil listhbas</command>
<command name="lputil-count">lputil count</command>

<!-- %s is replaced with the board index number: -->
```

User privileges and information access for Windows operating systems

Windows discovery notes
This section provides some information about discovering Windows hosts.

Potential user lock out
By default, AD accounts have a limited number of login attempts (for example, three attempts in fifteen minutes). Access Denied errors from WMI, DCOM, and RemQuery are counted as unsuccessful login attempts. Where target hosts are incorrectly configured, this limit can be exceeded and the account locked out.

To avoid this, configure the BMC Atrium Discovery account to accept unlimited login attempts.

Firewalls
Some versions of Windows have a default firewall configuration that does not permit discovery. You should configure the firewall to permit access; otherwise you will be unable to discovery your Windows hosts. See Discovery communications for information on the ports that should be open.

Windows Domain Controllers
In order to get a full set of data from a Windows system, the credential used has to be in the Local Administrator group for the target. Domain Controllers have the equivalent of a local administrator, however the local administrator on a Domain Controller has sufficient permissions to become a domain administrator. The implication of this being that having full local administration rights on the Domain Controller effectively means you have a Domain Admin account.

**Windows Server 2008 and later and Windows Vista and later**
The account being used to discover the target host must be one of the following:

- A domain user with Administrator privileges on the target host.
- A non-domain user with Administrator privileges and with remote UAC disabled on the target host.

**Windows 2000 and Windows NT**
RemQuery discovery uses AES encryption. This is not supported in Windows 2000 so RemQuery discovery falls back to DES encryption. Windows NT does not support AES or DES so RemQuery discovery is unencrypted. WMI discovery is unaffected.

![getServices method requires WMI](image)

**getServices method requires WMI**

In Windows 2000 and Windows NT the `sc.exe` executable is not provided. The `getServices` method requires WMI to run successfully.

**Windows discovery using IPv6**
Windows discovery using IPv6 is not supported for the following versions of Windows for the proxy host or the target host:

- Windows Server 2003
- Windows XP
- Windows 2000

To discover these versions of Windows, you must use IPv4.

Proxy pools can only contain proxies from one of the following groups:

- proxies running on the "IPv6 unsupported" versions of Windows noted above
- later versions where IPv6 is supported, such as Windows Server 2008, and Windows 7.

**Windows discovery commands**
The following table show the commands that are run on Windows platforms. The following methods are used:

- **WMI**: Windows proxies use Windows Management Instrumentation (WMI) as the primary means of discovery. Discovery uses both WMI queries and WMI registry access.
- **RemQuery**: If WMI does not succeed, the proxies use various command line tools via the RemQuery utility. When it is used, it is copied onto the admin$ share of the scanned host, installed and started as a service. The service is then used to execute the discovery scripts. At the end of the scan, the service is stopped and uninstalled, but the executable is left in the admin$ share. If a copy already exists, and is the same version, it is not copied again. If the copy is of an older version, it is updated.

- **SNMP**: SNMP discovery is supported for all devices with an accessible SNMP agent. Discovery supports SNMP v1, v2c and v3. For some older platforms (for example, Netware) the use of SNMP v1 might be required. This is defined on a per-credential basis. Only read (GET, GETNEXT, GETBULK) access is required.

---

### WMI

<table>
<thead>
<tr>
<th>Method</th>
<th>WMI Namespace</th>
<th>WMI Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo'</td>
<td>WMI Namespace</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>root\CIMV2</td>
<td>ASSOCIATORS OF (Win32_Directory='\path') WHERE ResultClass = CIM_LogicalFile</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_LogicalDisk WHERE DriveType = 3 or DriveType = 4</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_LogicalDiskToPartition</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_Share</td>
</tr>
<tr>
<td>getHBAInfo</td>
<td>root\WMI</td>
<td>SELECT * FROM MSFC_FCAdapterHBAAttributes</td>
</tr>
<tr>
<td></td>
<td>root\WMI</td>
<td>SELECT * FROM MSFC_FibrePortHBAAttributes</td>
</tr>
<tr>
<td>getHostInfo'</td>
<td>root\CIMV2</td>
<td>SELECT Name, Manufacturer, Model, Domain, SystemType FROM Win32_ComputerSystem</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>Optional, This query can fail.</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT DNSDomain FROM Win32_NetworkAdapterConfiguration WHERE IPEnabled = 1</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_OperatingSystem</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT SystemUpTime FROM Win32_PerfFormattedData_PerfOS_System</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT Capacity FROM Win32_PhysicalMemory</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT SerialNumber FROM Win32_BIOS</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT Vendor, IdentifyingNumber, Name, UUID FROM Win32_ComputerSystemProduct</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_Processor</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT HotFixID, ServicePackInEffect FROM Win32_QuickFixEngineering</td>
</tr>
<tr>
<td>Method Notes</td>
<td>WMI Namespace</td>
<td>WMI Query</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapterConfiguration</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapter</td>
</tr>
<tr>
<td>getMACAddresses</td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapterConfiguration</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapter</td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapterConfiguration</td>
</tr>
<tr>
<td></td>
<td>root\CIMV2</td>
<td>SELECT * FROM Win32_NetworkAdapter</td>
</tr>
<tr>
<td>getPackageList</td>
<td>See notes below.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>root\default:</td>
<td>HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall*\DisplayName</td>
</tr>
<tr>
<td></td>
<td>StdRegProv</td>
<td>root\default:</td>
</tr>
<tr>
<td></td>
<td>%key%</td>
<td>root\default:</td>
</tr>
<tr>
<td>getPatchList</td>
<td>Handled by getHostInfo call, specifically: SELECT HotFixID, ServicePackInEffect FROM Win32_QuickFixEngineering</td>
<td></td>
</tr>
<tr>
<td>getProcessList</td>
<td>Calls getOwner() on each WMI object returned.</td>
<td>root\CIMV2</td>
</tr>
<tr>
<td>getRegistryListing</td>
<td>Registry keys are passed directly to the standard registry provider.</td>
<td>root\default:</td>
</tr>
<tr>
<td></td>
<td>StdRegProv</td>
<td>root\default:</td>
</tr>
<tr>
<td>getRegistryValue</td>
<td>Registry values are passed directly to the standard registry provider.</td>
<td>root\default:</td>
</tr>
<tr>
<td></td>
<td>StdRegProv</td>
<td>root\CIMV2</td>
</tr>
</tbody>
</table>

*indicates methods that must succeed for a Host to be created

getPackageList
Package information is obtained by walking these registry keys described in the previous table rather than using Win32_Product, as it provides more reliable data.
In order to speed this process, a temporary WMI class is created on the remote computer to query the registry locally. This temporary class is given a unique name and is removed once the registry data has been retrieved.

On 64 bit Windows systems, the Wow6432Node (32 bit application data) is also examined.

**getHBAInfo**

WMI support for gathering HBA information uses the following queries to populate the HBA information if it is safe to do so:

```plaintext
SELECT * FROM MSFC_FCAdapterHBAAttributes
SELECT * FROM MSFC_FibrePortHBAAttributes
```

The OS version and patch list is checked to see whether HBA queries are safe. On Microsoft Windows Server 2003, Vista, and Server 2008 the HBAAPI.DLL module used by WMI leaks handles unless patched with KB957052. If this patch is not installed, no WMI requests are made.

By inspection, no current version of Windows 2003 (5.2.x) or Windows 2008 (6.0.x) has this patch included (current versions including service packs) but Windows 2008 R2 (6.1.x) has. It is not clear whether the problem exists on Windows 2000, though there is no patch available.

We make the following assumptions:

- Windows 2000 HBA queries are safe via WMI.
- Newer versions of Windows do not have the bug.
- This check is unnecessary when running FCINFO.EXE. This does use HBAAPI.DLL and could experience the same handle leak, but is a short lived process and they are cleared on exit.

The Microsoft FCINFO.EXE command line tool is also used by RemQuery. This is used where WMI is deemed unsafe (or has failed for some reason). This provides equivalent information about HBAs since it uses the same API as the WMI provider.

**RemQuery**

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>Handled by getHostInfo call.</td>
<td></td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>REMQUERY DIR /-C /TW /4 % path%</td>
<td></td>
</tr>
<tr>
<td>getFileContent</td>
<td>Handled by getFileInfo call.</td>
<td></td>
</tr>
<tr>
<td>getFileInfo</td>
<td>REMQUERY CMD /C DIR /-C /TW /4 % path%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REMQUERY CMD /C TYPE %path%</td>
<td></td>
</tr>
<tr>
<td>getFileMetadata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Script</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>getHBAInfo</td>
<td>REMQUERY FCINFO /DETAILS</td>
<td>Requires Microsoft FCINFO.EXE to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY HBACMD LISTHBS</td>
<td>Requires Emulex HBAnywhere to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY HBACMD HBAATTRIB %wwpn%</td>
<td>Requires Emulex HBAnywhere to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY LPUTIL LISTHBS</td>
<td>Requires Emulex LPUTIL.EXE to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY LPUTIL COUNT</td>
<td>Requires Emulex LPUTIL.EXE to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY LPUTIL FWLIST %board_id%</td>
<td>Requires Emulex LPUTIL.EXE to be installed on the target system.</td>
</tr>
<tr>
<td>getHostInfo*</td>
<td>REMQUERY WMIC BIOS GET SERIALNUMBER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REMQUERY WMIC CSPRODUCT GET UUID</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REMQUERY SYSTEMINFO /fo csv /nh</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REMQUERY &quot;HOSTNAME &amp;&amp; VER&quot;</td>
<td></td>
</tr>
<tr>
<td>getIPAddresses</td>
<td>REMQUERY</td>
<td>Uses Windows API to query IP addresses.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY IPCONFIG /ALL</td>
<td></td>
</tr>
<tr>
<td>getMACAddresses*</td>
<td>REMQUERY</td>
<td>Uses Windows API to query MAC addresses.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY IPCONFIG /ALL</td>
<td></td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>REMQUERY NETSTAT -ano</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REMQUERY NETSTAT -an</td>
<td></td>
</tr>
<tr>
<td>getNetworkInterfaces</td>
<td>REMQUERY</td>
<td>Uses Windows API to query interface details.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY IPCONFIG /ALL</td>
<td></td>
</tr>
<tr>
<td>getPackageList</td>
<td>REMQUERY</td>
<td>Uses Windows API to request same registry keys as WMI Queries.</td>
</tr>
<tr>
<td>getPatchList</td>
<td></td>
<td>Handled by getHostInfo call.</td>
</tr>
<tr>
<td>getProcessList</td>
<td>REMQUERY</td>
<td>Uses Windows API to query process information.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY TASKLIST /fo /csv /nh /v</td>
<td></td>
</tr>
</tbody>
</table>
### Method

<table>
<thead>
<tr>
<th>Method</th>
<th>Script</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>getProcessToConnectionMapping</code></td>
<td>REMQUERY TCPVCON -ano</td>
<td>Requires TCPVCON.EXE to be installed on the target system.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY OPENPORTS -netstat</td>
<td>Optional, must be enabled in the Proxy configuration. Requires OPENPORTS.EXE to be installed on the target system.</td>
</tr>
<tr>
<td><code>getRegistryListing</code></td>
<td>REMQUERY REG QUERY %hive%% key%</td>
<td></td>
</tr>
<tr>
<td><code>getRegistryValue</code></td>
<td>REMQUERY REG QUERY %hive%% key% /v %value%</td>
<td></td>
</tr>
<tr>
<td><code>getServices</code></td>
<td>REMQUERY</td>
<td>Uses Windows API to query process information.</td>
</tr>
<tr>
<td></td>
<td>REMQUERY SC QUERYEX state=all</td>
<td></td>
</tr>
</tbody>
</table>

* Indicates methods that must succeed for a Host to be created.

### SNMP

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>getDeviceInfo</code></td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.5.0</td>
</tr>
<tr>
<td></td>
<td>LanMgr-Mib-II-MIB::domPrimaryDomain.0</td>
<td>1.3.6.1.4.1.77.1.4.1.0</td>
</tr>
<tr>
<td><code>getHostInfo</code></td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td><code>getIPAddresses</code></td>
<td>IF-MIB::ifEntry [ifDescr, ifType, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1 [2.3.8] 1.3.6.1.2.1.4.34.1 [2.3.4.5]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddressEntry [ipAddressAddr, ipAddressIfIndex, ipAddressType, ipAddressPrefix]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipAddrEntry [ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetmask]</td>
<td>1.3.6.1.2.1.4.20.1 [.1.2.3] 1.3.6.1.2.1.55.1.8.1 [.1.2]</td>
</tr>
<tr>
<td></td>
<td>IPV6-MIB::ipv6AddrEntry [ipv6AddrAddress, ipv6AddrPrefixLength]</td>
<td></td>
</tr>
<tr>
<td><code>getMACAddresses</code></td>
<td>IF-MIB::ifEntry [ifDescr, ifType, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.4.20.1 [2.3.6.8]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToPhysicalEntry [ipNetToPhysicalAddr, ipNetToPhysicalAddress, ipNetToPhysicalType]</td>
<td>1.3.6.1.2.1.4.35.1 [.4.6]</td>
</tr>
<tr>
<td></td>
<td>IP-MIB::ipNetToMediaEntry [ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>1.3.6.1.2.1.4.22.1 [.2.4]</td>
</tr>
<tr>
<td><code>getNetworkConnectionList</code></td>
<td>TCP-MIB::tcpConnectionEntry [tcpConnectionLocalAddress, tcpConnectionLocalPort, tcpConnectionRemAddress, tcpConnectionRemPort, tcpConnectionState, tcpConnectionProcess]</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>MIB Values</td>
<td>OID</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>TCP-MIB::tcpListenerEntry</td>
<td>[tcpListenerLocalAddress, tcpListenerLocalPort, tcpListenerProcess]</td>
<td>1.3.6.1.2.1.6.19.1</td>
</tr>
<tr>
<td>UDP-MIB::udpEndpointEntry</td>
<td>[udpEndpointLocalAddress, udpEndpointLocalPort, udpEndpointProcess]</td>
<td>1.3.6.1.2.1.6.20.1</td>
</tr>
<tr>
<td>TCP-MIB::tcpConnEntry</td>
<td>[tcpConnState, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td>IPV6-TCP-MIB::ipv6TcpConnEntry</td>
<td>[ipv6TcpConnLocalAddress, ipv6TcpConnLocalPort, ipv6TcpConnRemAddress, ipv6TcpConnRemPort, ipv6TcpConnState]</td>
<td>1.3.6.1.2.1.6.16.1</td>
</tr>
<tr>
<td>UDP-MIB::udpConnEntry</td>
<td>[udpLocalAddress, udpLocalPort]</td>
<td>1.3.6.1.2.1.6.13.1</td>
</tr>
<tr>
<td>IP-V6-UDP-MIB::ipv6UdpEntry</td>
<td>[ipv6UdpLocalAddress, ipv6UdpLocalPort]</td>
<td>1.3.6.1.2.1.7.6.1</td>
</tr>
<tr>
<td>IF-MIB::ifEntry</td>
<td>[ifIndex, ifDescr, ifType, ifSpeed, ifPhysAddress, ifOperStatus]</td>
<td>1.3.6.1.2.1.2.2.1</td>
</tr>
<tr>
<td>IF-MIB::ifXEntry</td>
<td>[ifAlias, ifName, ifHighSpeed]</td>
<td>1.3.6.1.2.1.2.2.2</td>
</tr>
<tr>
<td>MAU-MIB::ifMauEntry</td>
<td>[ifMauIfIndex, ifMauType, ifMauAutoNegSupported]</td>
<td>1.3.6.1.2.1.2.2.3</td>
</tr>
<tr>
<td>EtherLike-MIB::dot3StatsEntry</td>
<td>[dot3StatsDuplexStatus]</td>
<td>1.3.6.1.2.1.2.2.4</td>
</tr>
<tr>
<td>IP-MIB::ipNetToPhysicalEntry</td>
<td>[ipNetToPhysicalIfIndex, ipNetToPhysicalPhysAddress, ipNetToPhysicalType]</td>
<td>1.3.6.1.2.1.2.2.5</td>
</tr>
<tr>
<td>IP-MIB::ipNetToMediaEntry</td>
<td>[ipNetToMediaIfIndex, ipNetToMediaPhysAddress, ipNetToMediaType]</td>
<td>1.3.6.1.2.1.2.2.6</td>
</tr>
<tr>
<td>HOST-RESOURCES-MIB::hrSWInstalledTable</td>
<td>[hrSWInstalledName]</td>
<td>1.3.6.1.2.1.2.2.7</td>
</tr>
<tr>
<td>HOST-RESOURCES-MIB::hrSWRunTable</td>
<td>[hrSWRunIndex, hrSWRunName, hrSWRunPath, hrSWRunParameters]</td>
<td>1.3.6.1.2.1.2.2.8</td>
</tr>
</tbody>
</table>

*indicates methods that must succeed for a Host to be created.

**Installing Windows proxies**

ℹ️ Proxy changes in version 10.1 and later
In BMC Atrium Discovery 10.1 and later, configuring secure communication with proxies has been improved with the new key and certificate management capabilities in the Windows proxy manager and with automatic registration of the proxy on the appliance. Upgraded proxies continue to use the legacy keys, but we recommend that you switch them to use unique keys as soon as all connected appliances have also been upgraded. For details, see Secure deployment.

Windows proxies and firewalls

The BMC Atrium Discovery appliance opens connections to the Windows proxies. The ports used for the proxies are chosen at installation time, and can be modified using the Proxy Manager. You must modify the proxy host firewall, and any other firewalls between the proxy host and the appliance to permit communication on the necessary ports.

Windows proxy installation

There can only be a single installation of the Windows discovery proxy on a host, which can manage and run multiple instances of the discovery proxy. Installing a newer Windows discovery proxy version will always upgrade all configured instances.

Before you begin

- Ensure that you have the necessary permissions to download and install the Windows proxies and Window proxy manager.
- Understand the Windows proxy version and Operating System (OS) compatibility to scan and retrieve IPv6 data.
- Ensure the minimum recommended specifications for the Windows proxy host.
- Consider the ports that must be open in any firewall between the appliance and the proxy or proxies, and the proxies and target hosts.

Permissions required

You can download the Windows proxies and Windows proxy manager as installation files from the appliance and install onto the local Windows host. To install Windows proxies:

- You must be logged in as an administrator. If the software is not installed as this user then you need to grant permissions to write to C:\Program Files\BMC Software\ADDM Proxy.
- The user that runs the Windows proxy must have necessary permissions to read from and write to the etc, log, and record directories.
As a user on the appliance, you must have been granted the `admin/software/slave/download` permission to download the Windows proxy installers.

**Windows proxy version and OS compatibility**

To discover IPv6 hosts, the OS of the target hosts must be compatible with the OS of the computer on which the proxy service is running and a suitable discovery method must be used. For more information, see Operating System compatibility.

The following table provides information about the compatibility between Windows proxy types and versions, and the operating systems that the Windows proxy runs on for BMC Atrium Discovery.

<table>
<thead>
<tr>
<th>Windows Proxy Type</th>
<th>Earliest Compatible Windows Proxy Version</th>
<th>Windows Proxy Available for Supported Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Windows proxy</td>
<td>8.2 — with update to default SSL keys.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only</td>
</tr>
<tr>
<td></td>
<td>9.0 — no actions required.</td>
<td>Windows 2008 - Service Pack 2 (x86 and x86_64)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2008 R2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012 R2</td>
</tr>
<tr>
<td>Active Directory Windows proxy</td>
<td>8.2 — with update to default SSL keys.</td>
<td>Windows 2003 SP2 (x86 and x86_64) IPv4 discovery only</td>
</tr>
<tr>
<td></td>
<td>9.0 — no actions required.</td>
<td>Windows 2008 - Service Pack 2 (x86 and x86_64)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2008 R2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2012 R2</td>
</tr>
</tbody>
</table>

⚠️ **Workgroup Windows proxy**

The Workgroup Windows proxy is no longer supported. Running the Active Directory Windows proxy under a Workgroup account provides exactly the same functionality as the old Workgroup Windows proxy.

**Minimum host specification**

The following are the minimum recommended specifications for the Windows proxy host:

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>As stated in tables above</td>
</tr>
<tr>
<td>CPU</td>
<td>2GHz Intel Pentium® 4 CPU 512k Cache (or equivalent from other manufacturer)</td>
</tr>
<tr>
<td>Memory</td>
<td>2GB</td>
</tr>
<tr>
<td>Hard disk</td>
<td>60GB</td>
</tr>
</tbody>
</table>
1. To avoid any impact during resource-intensive periods of discovery, it is strongly recommended not to install the Windows proxy on any host supporting other business services. This is true even if the minimum Windows proxy specification is exceeded, since the Windows proxy will attempt to use what resources are available, in order to optimize scan throughput.

Windows discovery communications

You should also consider the ports that will need to be opened in any firewall between the appliance and the proxy or proxies, and the proxies and target hosts.

Windows discovery metadata

Discovery metadata covers Windows as well as UNIX. This provides information about why sessions failed to be established and why scripts failed to run, including information about what credential or Windows proxy was used.

Downloading the Windows proxy installer

When you download the proxy installer from the appliance user interface (UI), you download a single installer file from which you install the following:

- Windows proxy manager
- Active Directory proxy
- Credential Windows proxy

To download the Windows proxy installer:

1. From the Tools section of the Discovery page, click the Download installer for Windows Proxy version 10.1.00 link.
2. Save the installation file to your file system.

Installing the Windows proxy manager and proxies

### Installing or upgrading Windows proxies where anti-virus software is installed

Before installing Windows discovery proxies you should either disable the anti-virus software or configure it to exclude RemQuery from triggering a virus alert. You can enable the anti-virus software once the Windows proxy has been installed.

To install the Windows proxy manager and Windows proxies:

1. Run the installer by double-clicking on the downloaded installer file. A welcome screen is displayed.
2. Click **Next**.
3. Click **Browse...** to select an installation directory, or click **Next** to accept the default installation directory (C:\Program Files\BMC Software\ADDM Proxy). The bottom of the Select Destination Location screen displays the minimum free disk space required in MB.
4. To create the Windows proxy application's shortcuts, click **Browse** to select a different folder, or click **Next** to accept the default folder (`BMC Software\ADDM Proxy`). If you choose **Don't create a Start Menu Folder** here, ensure that you clear all the start menu option check boxes in the next step.

5. On the Select Additional Tasks screen, select the options that will be available in the Start menu, and then click **Next**.

6. To install an Active Directory Proxy, select the Install Active Directory Proxy check box.
   a. Enter the credentials for the user account that will run the Windows proxy. If you do not enter the credentials at this point you can do so later, see **Specifying the Account Used to Run the Windows proxy**. The Windows proxy will run as the Local System user if credentials are not entered. However, an Active Directory Proxy running as a Local System user will not have the necessary domain credentials to perform any discovery.
   b. Click **Next**.

7. To install a Credential Proxy, select the Install Credential Proxy check box.
   a. Enter the credentials for the user account that will run the Windows proxy. You must prefix the user name with `localhost` (for example, `localhost\Administrator`). If you do not enter the credentials at this point you can do so later. The Windows proxy will run as the Local System user if credentials are not entered.
   b. Click **Next**.

8. Review the details in the Ready to Install window. If the details are incorrect, click **Back** and navigate through the installer to correct the error. If they are correct, click **Install** to install the selected components.

9. The Completing the BMC Atrium Discovery Proxy Setup Wizard is displayed.
   a. (If you have already installed the Active Directory proxies) To register the proxies with the appliance, select Register Active Directory Proxy with ADDM Appliance.
   b. (If you have already installed the Credential proxies) To register the proxies with the appliance, select Register Credential Proxy with ADDM Appliance.
   c. To run the Windows proxy manager immediately after installation, select Run Proxy Manager.

   The BMC Atrium Discovery UI Create Windows proxy page, pre-populated with details of this Windows proxy is displayed when this part of the setup is complete.
The Create Windows proxy page is populated with the certificate of the proxy. This certificate is used for securing communications between the appliance and the proxy. You can verify that the proxy communication has not been intercepted by comparing the certificate fingerprint shown in the appliance UI with the one shown in the Proxy Manager's Key And Certificate Management dialog.

10. To exit the installer, click **Finish**.

### Service startup failure

Sometimes Windows might refuse the installer permission to start the Windows proxy service, resulting in a dialog box along the lines of *service installed but could not be started*. This is remedied by manually supplying the credentials directly to the service using the Windows Services control panel. See *Specifying the Account Used to Run the Windows proxy*.

###Registering a Windows Proxy from the appliance UI

If the proxy is not able to register with the appliance automatically (due to connectivity issues or to strict security policies), you can instead register the proxy using the appliance UI. When registered this way, the connection from the appliance must be approved in the Known Appliances dialog of the Proxy Manager.

#### Post installation settings

The following sections detail post installation settings and modifications that might be required for Windows proxies.

**To modify the Windows proxy host firewall**

By default, the Windows firewall blocks the ports that the Windows proxies use. To enable an appliance to communicate with a Windows proxy, you must amend the firewall rules to permit communication on the ports that each Windows proxy type installed is using. The Proxy Manager displays the port that each proxy is using.

To modify the host firewall, select **Windows Firewall** from the Windows Control Panel. You can add a Windows proxy as an exception (as a program or a port) on the exceptions tab.
To specify the account used to run the Windows proxy

The Active Directory Windows proxies gain their permissions on the discovery target from the user account that they run as, whereas the Credential proxies gain their permissions on the discovery target from the credentials entered in Discovery > Credentials > Devices > Hosts. The recommended procedure to configure or edit the account used to run the Windows proxy is from the Windows proxy manager. For more information, see specify user account and edit user account.

The alternative method to configure the account used to run the Windows proxy is as follows:

1. Choose Start > Settings > Control Panel.
2. Double-click Administrative Tools and then Services.
3. Right-click the Windows proxy entry in the Services list and choose Properties.
4. Switch to the Log On tab and select This account.
5. Depending on the proxy type to be configured, perform the following:
   - For a Credential Proxy, enter the user name and password for a valid local user account, which should be in the local Administrators group.
   - For an Active Directory proxy, enter the user name and password of the Domain account that the service is to run as.
     You may see a dialog saying that the user has been granted the Log on as a Service right.
6. To apply the changes, click OK.

To start or stop the Windows proxy

The recommended procedure to start or stop the Windows proxy is from the Windows proxy manager. For more information, see start and stop proxy.

The alternative method to start or stop a Windows proxy is as follows:
From the Control Panel, navigate to Administrative Tools and access the Services list. Select the Windows proxy that you want to start or stop. The services panel is refreshed with information and links enabling you to start, restart, or stop the Windows proxy.

1. To start the Windows proxy, click Start the service.
2. To restart the Windows proxy, click Restart the service.
3. To stop the Windows proxy, click Stop the service.

To start the Windows proxy automatically

When you create a new proxy from the Windows proxy manager, you can configure the proxy to start automatically. For more information, see create a new proxy.

The alternative method to set the Windows proxy to start automatically is as follows:

1. From the Control Panel, navigate to Administrative Tools and access the Services list.
2. Select the Windows proxy that you want to start automatically and select Properties from the popup menu.

3. Select **Automatic** from the Startup type: drop-down list, and click **OK**.

**To specify additional startup options**

When the Windows proxy is working satisfactorily, you can specify additional startup options. Many of these can also be specified through the **Manage Windows Proxy page**. These are noted in the following table.

To address unusual requirements, you can also enter the start up options described in the table in the registry key appropriate for the Windows proxy type and host architecture. On a 32 bit system this is one of:

- **Active Directory Windows proxy**
  
  HKEY_LOCAL_MACHINE\SOFTWARE\BMC Software\Atrium Discovery\ADPROXY\CommandLine

- **Credential Windows proxy**
  
  HKEY_LOCAL_MACHINE\SOFTWARE\BMC Software\Atrium Discovery\PROXY\CommandLine

On a 64 bit system this is one of:

- **Active Directory Windows proxy**
  
  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\BMC Software\Atrium Discovery\ADPROXY\CommandLine

- **Credential Windows proxy**
  
  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\BMC Software\Atrium Discovery\PROXY\CommandLine

**Option** | **Description**
--- | ---
--auto-purge-all | You can configure the Windows proxy to automatically purge its log and record data directories. The default behavior is not to purge. Specifies that log and record data directories will be purged. Set via UI.

--auto-purge-logs | Specifies that only log directories will be purged. Set via UI.

--auto-purge-record | Specifies that only record directories will be purged. Set via UI.

--auto-purge-max-data-age value | Specify an age above which data is automatically purged. This is set in days and the default is seven. Set via UI.

--auto-purge-period value | The frequency at which the automatic purge occurs. This is set in hours and the default is 24 (daily). Set via UI.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--log-soft-limit value</td>
<td>A size limit (in MB) for the log directories. If this limit is exceeded the oldest records will be deleted. The default behavior is not to specify a limit (zero).</td>
</tr>
<tr>
<td>--record-soft-limit value</td>
<td>A size limit (in MB) for the record data directories. If this limit is exceeded the oldest data will be deleted. The default behavior is not to specify a limit (zero).</td>
</tr>
<tr>
<td>--enable-config-upload</td>
<td>Enable or disable uploading configuration, overriding the setting specified in the configuration file.</td>
</tr>
<tr>
<td>--disable-config-upload</td>
<td></td>
</tr>
<tr>
<td>--config-file-limit value</td>
<td>The number of backup configuration files to keep. The default is none. If this is exceeded, the oldest file is deleted.</td>
</tr>
<tr>
<td>--conf &lt;config file&gt;</td>
<td>Specify a configuration file to use.</td>
</tr>
<tr>
<td>--openports</td>
<td>Enable or disable OpenPorts, overriding the setting specified in the configuration file.</td>
</tr>
<tr>
<td>--no-openports</td>
<td></td>
</tr>
<tr>
<td>--tcpvcon</td>
<td>Enable or disable Tcpvcon, overriding the setting specified in the configuration file.</td>
</tr>
<tr>
<td>--no-tcpvcon</td>
<td></td>
</tr>
<tr>
<td>--dont-resolve-hostnames</td>
<td>The getInfo method retrieves patch, device, and host information. If no hostname is found then a reverse DNS lookup is performed to determine the hostname. Specify --dont-resolve-hostnames to prevent this.</td>
</tr>
<tr>
<td>--remquery</td>
<td>Enable or disable RemQuery, overriding the setting specified in the configuration file. Set through the user interface (UI).</td>
</tr>
<tr>
<td>--no-remquery</td>
<td></td>
</tr>
<tr>
<td>--remquery-timeout value</td>
<td>Specify a timeout value (in seconds) for RemQuery calls. The default is 60 seconds. Set through the UI.</td>
</tr>
<tr>
<td>--wmi</td>
<td>Enable or disable WMI, overriding the setting specified in the configuration file. Set through the UI.</td>
</tr>
<tr>
<td>--no-wmi</td>
<td></td>
</tr>
<tr>
<td>--wmi-timeout value</td>
<td>Specify a timeout value (in seconds) for WMI queries. The default is 120 seconds. Set through UI.</td>
</tr>
</tbody>
</table>

To test Windows credentials and communication

You can test the credentials by using it to discover a Windows computer that you know the user can access. To do this, from a command prompt on the Windows proxy, use the `runas` command to run a Discovery command such as `systeminfo` as the Domain user:

```
C:\> runas /user:DOMAIN\username "systeminfo /S TARGET"
```

Replace `DOMAIN` with the domain name, for example `TIDEWAY`, `username` with the user name, for example `discovery`, and `TARGET` with the resolvable hostname or IP address.
Upgrading a Windows proxy

Before upgrading you must ensure that existing Windows proxies are not running. If you do not do this, the install will fail and you will need to reboot the computer.

The upgrade process installs a new proxy and configures it using information taken from the previous proxy. It then uninstalls the previous proxy. As a consequence of this sequence:

- **Upgrading from versions prior to 8.3 SP2** — You should not use the same installation directory as the existing Windows proxy or the installation will fail. Accepting the upgrade default prevents this.
- **Upgrading a Workgroup proxy** — The Windows Workgroup proxy is deprecated. If you upgrade a Workgroup proxy it is converted into an AD proxy. See [Windows proxy compatibility matrix](#) for more information.

### Proxy username/password and upgrading

During the upgrade process you need to enter the Active Directory credentials. Usernames are preserved during the upgrade, but passwords are not.

### Running the upgrade

To install the Windows proxy manager and upgrade all Windows proxies:

1. Run the installer by double-clicking on the downloaded installer file. A welcome screen is displayed.
2. Click **Next**.
3. Click **Browse** to select the installation directory, or click **Next** to accept the default default installation directory (C:\Program Files\BMC Software\ADDM Proxy). See the upgrade notes above.
4. To create the Windows proxy application’s shortcuts, click **Browse** to select a different folder, or click **Next** to accept the default folder (BMC Software\ADDM Proxy). If you choose **Don’t create a Start Menu Folder** here, ensure that you clear all the start menu option check boxes in the next step.
5. On the Select Additional Tasks screen, choose options that will be available in the Start menu, and then click **Next**.
6. If an existing Active Directory proxy is found, you are asked to enter credentials. The Account field is pre-populated with the username that the proxy is using. Enter the corresponding password in the Password field. If the proxy is currently running as the local system user, you are prompted for an Active Directory username and password. Although the proxy service will run as the local system user, it will not be able to perform any discovery activities until it is run as an Active Directory user. Select Migrate existing configuration file to migrate any custom changes to the new proxy.
7. Click **Next**.
8. If an existing Workgroup proxy is found, it is converted to an Active Directory proxy. You are asked to enter credentials. The Account field is pre-populated with the username that the proxy is using. Enter the corresponding password in the Password field. If the proxy is currently running as the local system user, you are prompted for an username and password. Although the proxy service will run as the local system user, it will not be able to perform any discovery activities until it is run as a Workgroup user. Select Migrate existing configuration file to migrate any custom changes to the new proxy.

9. Click Next.

10. If an existing Credential proxy is found, the default is not to request credentials. BMC recommends that credential proxies are run as local named users with administrator privileges, not the local system user. Select Migrate existing configuration file to migrate any custom changes to the new proxy.

11. Click Next.

12. Review the details in the Ready to Install window. If the details are incorrect, click Back and navigate through the installer to correct the error. If they are correct, click Install to install the selected components.

13. You are asked whether you want to keep the existing configuration for each proxy. Click Yes to keep the configuration.

14. Check the Run Proxy Manager check box to run the Windows proxy manager immediately after installation.

15. To exit the installer, click Finish.

**Silent installation**

The Windows proxy manager and proxy installer uses Inno Setup which provides silent installation capabilities at the command line.

To invoke the installer at the command line:

1. Using a command prompt, change directory to the directory into which you downloaded the installer file. Enter:

   ```
   C:\>cd "Documents and Settings\username\My Documents\Download\"
   ```

2. Run the installer using the Inno Setup options and the additional Windows proxy manager installer options. Enter:

   ```
   addmproxy_installer_10.1_xxxxxx.exe /SILENT /ADUSER="username" /ADPASSWORD="password"
   ```

   The Inno Setup options are described on their [website](#).

   Additional Windows proxy manager and proxy installer options are described in the following table:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ADCREATE=Y</td>
<td>N</td>
</tr>
<tr>
<td>/CREDCREATE=Y</td>
<td>N</td>
</tr>
<tr>
<td>/ADUSER=&quot;username&quot;</td>
<td>The username with which to run an AD proxy. The default is &quot;&quot;.</td>
</tr>
<tr>
<td>/ADPASSWORD=&quot;password&quot;</td>
<td>The corresponding password. The default is &quot;&quot;.</td>
</tr>
<tr>
<td>/CREDU&quot;username&quot;</td>
<td>The username to run a credential proxy. The default is &quot;&quot;.</td>
</tr>
<tr>
<td>/CREDPASSWORD=&quot;password&quot;</td>
<td>The corresponding password. The default is &quot;&quot;.</td>
</tr>
<tr>
<td>/SILENT</td>
<td>Run a BMC Atrium Discovery Proxy silent installation. For installing AD proxy, the /ADUSER and /ADPASSWORD are mandatory. For installing just a Credentials proxy, Active Directory credentials are not required, as the installer uses system account credentials.</td>
</tr>
</tbody>
</table>

These commands are entered as a space separated list.

For silent installation of the Active Directory proxies, you must provide valid domain credentials. The installation process verifies this, and it fails if default username and password was provided.

**Windows proxy downgrade**

If you need to downgrade a Windows proxy, you *must* stop the Windows proxy, uninstall it, and then install the new Windows proxy according to the instructions for that Windows proxy version.

**Windows proxy manager**

The Windows proxy manager enables you to install and manage proxies on the Windows host on which the manager is installed. The Windows proxy manager is installed when you install a proxy. For more information, see [Installing Windows proxies](#).

**To run the Windows proxy manager**

Select **Start > Programs > BMC Software > ADDM Proxy > Proxy Manager**.

The Windows proxy manager is displayed with a credential and an Active Directory proxy, as shown in the following illustration.

**Managing proxies**

You can perform the following proxy management tasks from the Windows proxy manager:
• Create (install a new proxy service)
• Edit the port and the user account that the proxy uses
• Delete (uninstall a proxy service)
• View the log directory for a proxy
• Start a selected proxy
• Stop a selected proxy
• Restart a selected proxy

These are described in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Button /Menu</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a new proxy</td>
<td>Create menu</td>
<td>Click the Create button, or from the corresponding menu:</td>
</tr>
<tr>
<td></td>
<td>/Menu</td>
<td>1. Enter the following information on the create proxy dialog:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Name</strong> — A name for the proxy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Type</strong> — Select either Active Directory or Credential.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Port</strong> — Enter a port if you need to use a specific unused port. Otherwise, choose the port which is automatically populated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Log on as</strong> — Select either Local System or This account. If you select Local System, the Windows proxy will run as the Local System user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you select This account, you must enter the domain and username for the user account that will run the Windows proxy in the format and the corresponding password.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Options</strong> — Select the check box to run the proxy immediately after installation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Register With</strong> — Select the appliance for the proxy to register.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. To create the new proxy, click <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You are prompted to login to the Create Windows proxy page of the UI with pre-populated proxy details.</td>
</tr>
</tbody>
</table>

_i Proxy registration using IPv6_

When registering a proxy with an appliance using IPv6, the proxy IP address that is automatically populated in the Create Windows Proxy page might be a temporary address. The appliance can communicate while the temporary address is valid, but when no longer valid, communication is not possible. This behavior has only been observed in Internet Explorer.

<table>
<thead>
<tr>
<th>Task</th>
<th>Button /Menu</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit the selected proxy</td>
<td>Create menu</td>
<td>Click the Edit button.</td>
</tr>
<tr>
<td></td>
<td>/Menu</td>
<td>You can edit the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Port that the proxy uses</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• User account that the proxy runs as</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. When prompted, confirm the changes by clicking <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Changes in the user account automatically updates the file permissions and restart the proxy service.</td>
</tr>
<tr>
<td>Delete the selected proxy</td>
<td>Create menu</td>
<td>Click the Delete button.</td>
</tr>
<tr>
<td></td>
<td>/Menu</td>
<td>2. When prompted, click <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Task</td>
<td>Button /Menu</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Open Log Directory</td>
<td><img src="image" alt="Alternate text." /></td>
<td>Click the Open Log Directory button. The log directory for the currently selected proxy is displayed in an Explorer window (an example pathname is C:\Program Files\BMC Software\ADDM Proxy\runtime\AD4\log).</td>
</tr>
<tr>
<td>Start</td>
<td><img src="image" alt="Alternate text." /></td>
<td>Click the start button. When the proxy starts, a green check icon is displayed next to its name.</td>
</tr>
<tr>
<td>Stop</td>
<td><img src="image" alt="Alternate text." /></td>
<td>Click the stop button. When the proxy stops, a red cross icon is displayed next to its name.</td>
</tr>
<tr>
<td>Restart</td>
<td><img src="image" alt="Alternate text." /></td>
<td>Click the restart button to stop and start the proxy.</td>
</tr>
<tr>
<td>Refresh proxy list</td>
<td><img src="image" alt="Alternate text." /></td>
<td>Click the refresh button to refresh the list. Sometimes the list does not pick up changes in the status of services, so it is good practice to refresh the list before starting or stopping a proxy.</td>
</tr>
</tbody>
</table>

⚠️ Editing proxy config files is only undertaken through the main user interface

You cannot edit the proxy configuration file as that is better handled centrally from the appliance.

Managing known appliances

🔍 Proxy needs the appliance certificate to enable communication

To enable communication between the proxy and the known appliance, the proxy must have the certificate of the appliance. The certificate can be requested from the add appliance dialog, or the appliance will automatically transfer its certificate for approval.

You can manage the appliances that are permitted to connect to the proxies from the Windows proxy manager.

To manage known appliances, click Edit > Known Appliances. The Known Appliances dialog is displayed which enables you to perform the following tasks:

- View the known appliances
- Add new appliances
- Approve appliances that wish to communicate with the proxy
• Edit known appliances
• Delete known appliances

These are described in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View the known appliances</td>
<td>The Known Appliances dialog displays the available known appliances.</td>
</tr>
<tr>
<td>Add new appliance</td>
<td>1. From the Known Appliances dialog, click the Add a new Appliance address button.</td>
</tr>
<tr>
<td></td>
<td>2. Enter the following information on the Add New Appliance dialog:</td>
</tr>
<tr>
<td></td>
<td>• Address: IP address or DNS name of the new appliance to be added.</td>
</tr>
<tr>
<td></td>
<td>3. Hit enter or click <strong>Connect</strong>.</td>
</tr>
<tr>
<td></td>
<td>4. To register with the new appliance, click <strong>Register</strong>.</td>
</tr>
<tr>
<td>Approve appliance</td>
<td>Unknown appliances that have contacted the proxy are listed in orange, with a status of <strong>Pending Approval</strong>.</td>
</tr>
<tr>
<td></td>
<td>1. From the Known Appliances dialog, click the appliance to approve</td>
</tr>
<tr>
<td></td>
<td>2. Click the <strong>Edit</strong> button.</td>
</tr>
<tr>
<td></td>
<td>3. Check the certificate fingerprint, and if the appliance should be approved, click <strong>Approve</strong>.</td>
</tr>
<tr>
<td></td>
<td>If the appliance should not be permitted to connect to the proxy, it may be deleted from the Known Appliances.  However, appliances periodically ping the proxies they are configured to use, so unless the appliance in question has been reconfigured to stop trying to use the proxy, the <strong>Pending Approval</strong> entry will soon be back.</td>
</tr>
<tr>
<td>Edit known appliance</td>
<td>1. From the Known Appliances dialog, click the appliance to edit.</td>
</tr>
<tr>
<td></td>
<td>2. Click the <strong>Edit</strong> button.</td>
</tr>
<tr>
<td></td>
<td>You can edit the address or name of the appliance.</td>
</tr>
<tr>
<td></td>
<td>In exotic circumstances, you can also manually enter a CA certificate. Click <strong>Edit CA Certificate</strong>, insert a valid CA Certificate for the selected known appliance, and click <strong>Ok</strong>.</td>
</tr>
<tr>
<td></td>
<td>3. To save the changes, click <strong>Ok</strong>.</td>
</tr>
<tr>
<td>Delete known appliance</td>
<td>1. From the Known Appliances dialog, click the appliance to delete.</td>
</tr>
<tr>
<td></td>
<td>2. Click the <strong>Delete the selected Appliance address</strong> button.</td>
</tr>
<tr>
<td></td>
<td>3. When prompted, confirm the action by clicking <strong>Yes</strong>.</td>
</tr>
</tbody>
</table>

**Managing keys and certificates**

To manage the proxies' keys and certificates, click **Edit > Key and Certificate Management**. The Key and Certificate Management dialog is displayed which enables you to perform the following tasks:

• View the Proxy Key Fingerprint
• View the Proxy Certificate Fingerprint
• Generate a new key and certificate
• Install the legacy key and certificate
• Show the certificate

These are described in the following table:
# Task Description

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View the Proxy Key Fingerprint</td>
<td>The <strong>Key and Certificate Management</strong> dialog displays Proxy SSL Key Fingerprint.</td>
</tr>
<tr>
<td>View the Proxy Certificate Fingerprint</td>
<td>The <strong>Key and Certificate Management</strong> dialog displays Proxy CA Fingerprint.</td>
</tr>
<tr>
<td>Generate new key and certificate</td>
<td>Generates a new key and certificate</td>
</tr>
<tr>
<td></td>
<td>1. From the <strong>Key and Certificate Management</strong> dialog, click <strong>Generate new key</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. When prompted, confirm the action by clicking <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Notes:</td>
<td>This operation resets communication with all appliances. You will need to register with these appliances again.</td>
</tr>
<tr>
<td>Stop using legacy key</td>
<td>If the proxy is currently using the legacy key and certificate, switch it to use unique keys.</td>
</tr>
<tr>
<td></td>
<td>1. From the <strong>Key and Certificate Management</strong> dialog, click <strong>Stop using legacy key</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. When prompted, confirm the action by clicking <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Notes:</td>
<td><strong>Note</strong>: Appliances running BMC Atrium Discovery version 10.1 and later will automatically start accepting the new certificate; older appliances will no longer be able to communicate.</td>
</tr>
<tr>
<td>Install legacy key</td>
<td>Switches the proxies to use the legacy key that is used in BMC Atrium Discovery version 10.0 and earlier, so the proxy can be used with those old versions.</td>
</tr>
<tr>
<td></td>
<td>1. From the <strong>Key and Certificate Management</strong> dialog, click <strong>Install legacy key</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. When prompted, confirm the action by clicking <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Notes:</td>
<td><strong>Note</strong>: This operation resets communication with all appliances of version 10.1. You will need to register with these appliances again.</td>
</tr>
<tr>
<td>Show CA Certificate</td>
<td>From the <strong>Key and Certificate Management</strong> dialog, click <strong>Show Certificate</strong>.</td>
</tr>
<tr>
<td></td>
<td>The Certificate dialog displays the certificate in a portable format (pem) that might be used for transferring the certificate to the appliance manually.</td>
</tr>
</tbody>
</table>

## Additional tasks

You can perform the following additional tasks from the Windows proxy manager:

<table>
<thead>
<tr>
<th>Task</th>
<th>Menu</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh Windows proxy list</td>
<td>View &gt; Refresh</td>
<td>Refreshes the Windows proxy list displayed on the Windows proxy manager.</td>
</tr>
<tr>
<td>Open the Service Control Manager</td>
<td>View &gt; Service Control Manager</td>
<td>Opens the Service Control Manager (SCM).</td>
</tr>
<tr>
<td>View documentation</td>
<td>Help &gt; Documentation</td>
<td>Displays the Windows proxy manager online documentation.</td>
</tr>
<tr>
<td>View Windows proxy manager version</td>
<td>Help &gt; About</td>
<td>Displays the version of the Windows proxy manager.</td>
</tr>
<tr>
<td>Exit the Windows proxy manager</td>
<td>Proxy &gt; Exit</td>
<td>Exits the Windows proxy manager.</td>
</tr>
</tbody>
</table>
The Windows proxy manager enables you to perform only the tasks listed on this page. You can perform tasks like adding a Windows proxy to a pool, or configuring advanced proxy settings (for example, configuring the log level, recording mode, discovery methods, purging, and so on) only from the appliance's UI. To learn how to manage the Windows proxies from the appliance's UI, see **Managing Windows proxies**.

**Managing Windows proxy pools**

You can manage Windows proxy pools using the Windows proxy management page. This page displays the list of the available proxy pools and the corresponding proxies in each pool.

To navigate to the Windows proxy management page:

1. From the secondary navigation bar on the **Discovery** tab, click **Credentials**.
2. Click **Windows Proxies**.
   
The Device Credentials page for Windows proxies (Windows proxy management page) is displayed.

Each proxy pool contains either Credential proxies or Active Directory proxies. A proxy pool cannot contain proxies of both the types.

Based on the proxy version and version of the Operating System (OS) the proxy runs on, the pool capability is one of the following:

- **Fully IPv6 capable**: Can scan IPv6 addresses and retrieve IPv6 data. Where BMC Atrium Discovery version 9.0 or later proxies are running on Windows 2008 or later.
- **Cannot scan IPv6 addresses**: Can retrieve IPv6 data but the Windows version does not support scanning IPv6 addresses. Where BMC Atrium Discovery version 9.0 or later proxies are running on versions of Windows older than Windows 2008.
- **Not IPv6 capable**: Cannot scan IPv6 addresses and cannot retrieve IPv6 data. BMC Atrium Discovery proxies from versions older than 9.0.

It is possible to have proxies of different versions (BMC Atrium Discovery version 9.0 or earlier versions) in a single proxy pool, though this is not recommended. However, if a proxy does not have all the capabilities of the pool, it is disabled. For example if you add a pre-version 9.0 proxy to an IPv6 capable pool it is disabled. You can re-enable it once the proxy is upgraded. This enables you to upgrade all the proxies in a pool over a period of time rather than having to upgrade them all at the same time.

The Windows proxy management page can contain multiple Windows proxy pools. For more information about the Windows proxy pool list, see **Viewing Windows proxy pools**.

The minimum version and release options can be configured in the **Discovery configuration settings**.
Adding a Windows proxy pool

After you have installed the Windows proxies, you must add the Windows proxy pools to the appliance. Adding proxy pools enables the appliance to organize sets of Windows proxies together, balance the work load of the proxies for discovery tasks, and use them for discovering Windows devices.

The appliance and the Windows proxies communicate with each other using TLS. See system communications for more information.

To add a proxy pool, perform the following steps on the Windows proxy management page:

1. Click Add....
   The Create Windows Proxy Pool page is displayed.
2. Enter the following information about the new Windows proxy pool:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name that the Windows proxy pool will be referred to in the user interface. This name must be unique. The system validates it to be unique.</td>
</tr>
</tbody>
</table>

Matching Criteria

Select "Match All" to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:

- IPv4 address: for example 192.168.1.100.
- IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.
- IPv6 address: for example fda8:7554:2721:a8b3::3.
- IPv6 network prefix: for example fda8:7554:2721:a8b3::/64.

The following address types cannot be specified

- IPv6 link local addresses (prefix fe80::/64)
- IPv6 multicast addresses (prefix ff00::/8)
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

Warning: You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view. Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

Pills are not currently supported in Opera.
2. **Type**
   Select the type of the Windows proxy pool. A Windows proxy pool can contain only one type of Windows proxy. The available types are:
   - Windows Active Directory Proxies
   - Windows Credential Proxies
   - Windows Workgroup Proxies

   **Warning**: The Workgroup Windows proxy has not been supplied since before BMC Atrium Discovery version 8.2. All of its functionality has been moved into the Active Directory Windows proxy.

3. **Domains**
   This field is enabled only for the Active Directory proxy pools. The domain or a space-separated list of domains that the Windows proxy pool will discover. This specifies that the pool is preferred for the discovery of these domains.

4. **Description**
   A free text description of the Windows proxy pool.

---

3. Click **Apply**.
   The Windows proxy pool is added to the appliance.

You can add multiple Windows proxy pools to the appliance.

### Viewing Windows proxy pools

The Windows proxy management page displays the available Windows proxy pools and the Windows proxies in each pool. The Windows proxy pools are displayed in the order (from top to bottom) in which you have added them to the appliance.

You can change the order of the Windows proxy pools by the following ways:

- Using the colored margin at the left of the row, drag and drop the row to the required position in the user interface.
- To move a Windows proxy pool to the top or bottom of the list, click the **Actions** list and select **Move to top** or **Move to bottom**.

**Warning**: Reordering of Windows proxies in a Windows proxy pool is not supported, as the proxy used is selected automatically.

For each Windows proxy pool, the following fields are displayed:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the Windows proxy pool.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the Windows proxy pool (for example, Active Directory).</td>
</tr>
<tr>
<td>State</td>
<td></td>
</tr>
</tbody>
</table>
Based on the version of the proxies you have added to a pool and the version of the Operating System (OS) the proxies run on, a proxy pool displays one of the following states in brackets next to the Type field:

- Fully IPv6 capable: Contains proxies for version 9.0 or later and runs on Windows 2008 or later Operating System.
- Cannot scan IPv6 addresses: Contains proxies for version 9.0 or later and runs on an Operating System earlier than Windows 2008.

⚠️ From a Fully IPv6 capable pool, if you delete the latest version proxies and add proxies of a version earlier than 9.0, the pool state (Fully IPv6 capable) does not change. However, the newly added proxies in the pool are disabled.

**Domains**

This field is available for Active Directory Windows proxy pools only. It displays the applicable domains.

**Description**

A free text description of the proxy pool.

**Actions**

A list with the following options:

- Add Windows proxy: Enables you to add Windows proxies to the Windows proxy pool. For more information, see Managing Windows proxies.
- Manage: Opens the Edit Windows Proxy Pool page. For more information, see Editing Windows proxy pools.
- Delete: Enables you to delete the Windows proxy pool.
- Move to top: Moves the Windows proxy pool to the top of the list.
- Move to bottom: Moves the Windows proxy pool to the bottom of the list.

---

**Editing Windows proxy pools**

You can edit the proxy pools from the Edit Windows proxy pool page.

To edit a Windows proxy pool, perform the following steps from the Windows proxy management page:

1. Corresponding to the proxy pool you want to manage, click **Actions** and select **Manage**.

   The Edit Windows proxy pool page is displayed and contains the following sections:
   - Edit Windows Proxy Pool
   - Pool contents

2. The Edit Windows Proxy Pool section enables you to change the IP address range and domains associated with the Windows proxy pool as follows:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching Criteria</td>
<td>To enable the proxies to scan all the IP addresses associated with the pool, select Match All. By default, Match All is displayed as selected. To enable the proxies to scan specific IP addresses or IP ranges, you must clear Match All and enter the IP addresses information in one of the following formats:</td>
</tr>
</tbody>
</table>
2. Field name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IPv4 address (for example 192.168.1.100).</td>
</tr>
<tr>
<td>• IPv4 range (for example 192.168.1.100–105, 192.168.1.100/24, or 192.168.1.*)</td>
</tr>
<tr>
<td>• IPv6 network prefix (for example fda8:7554:2721:a8b3::/64).</td>
</tr>
</tbody>
</table>

3. The following address types cannot be specified

- IPv6 link local addresses (prefix fe80::/64)
- IPv6 multicast addresses (prefix ff00::/8)
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

### Type

Displays the type of the Windows proxy pool.

### Domains

The field is enabled only for the Active Directory proxy pools and is dimmed for Credential or Workgroup proxy pools. The domain or a space-separated list of domains that the Windows proxy pool will discover. This specifies that the pool is preferred for the discovery of these domains.

### Description

A free text description of the proxy pool.

3. To apply any changes that you have made in the Edit Windows Proxy Pool section, click **Apply**.

4. The Pool contents section displays the following information on the contents of the Windows proxy pool:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the Windows proxy.</td>
</tr>
<tr>
<td>Status</td>
<td>Displays whether the proxy is active or not and the IPv6 capability of the proxy. The IPv6 capability is displayed as one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Can get IPv6 data and scan IPv6 addresses: BMC Atrium Discovery version 9.0 or later proxies that run on Windows 2008 or later.</td>
</tr>
<tr>
<td>Field name</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>• Can get IPv6 data but Windows version does not support scanning IPv6 addresses: BMC Atrium Discovery version 9.0 or later proxies that run on a version of Windows earlier than Windows 2008.</td>
<td></td>
</tr>
<tr>
<td>• IPv6 not supported: Proxy version earlier than BMC Atrium Discovery version 9.0 proxies.</td>
<td></td>
</tr>
<tr>
<td>Version</td>
<td>Displays the version number of the Windows proxy.</td>
</tr>
<tr>
<td>Actions</td>
<td>To delete the Windows proxy, click Delete.</td>
</tr>
<tr>
<td></td>
<td>To ping the Windows proxy, click Ping.</td>
</tr>
</tbody>
</table>

If you want to add a new Windows proxy to this pool, click **Add**. For more information about adding a Windows proxy to a pool, see **Managing Windows proxies**.

**Managing Windows proxies**

Depending on the management task, you manage Windows proxies by the following ways:

- Using the Windows proxy manager
- Using the Windows proxy management page

**Managing proxies using the Windows proxy manager**

The Windows proxy manager is a simple tool which enables you to install and manage proxies on the Windows host on which the manager is installed. You can perform the following tasks from the Windows proxy manager:

- Install a new proxy service
- Register a proxy with an appliance
- Edit the port that the proxy uses and the user account that the proxy runs as
- Uninstall a proxy service
- Start a selected proxy
- Stop a selected proxy
- Restart a selected proxy

The Windows proxy manager is installed when you install a proxy. For more information, see **Windows proxy manager**.

**Managing Windows proxies from the appliance UI**

The BMC Atrium Discovery UI enables you to manage the Windows proxies in the following ways:

- Add Windows proxies to the pools
- View Windows proxies
- Edit Windows proxies
- Test Windows proxies

You can manage Windows discovery using the Windows proxy management page. This page displays the list of the available proxy pools and the corresponding proxies in each pool.
To navigate to the Windows proxy management page:

1. From the secondary navigation bar on the Discovery tab, click Credentials.
2. Click Windows Proxies.
   The Device Credentials page for Windows proxies (Windows proxy management page) is displayed.

Adding Windows proxies

Adding Windows proxies to the appliance enables the appliance to use them for discovering Windows devices. The appliance and the Windows proxies communicate with each other using TLS. For more information, see Secure deployment.

After installing the Windows proxies on the appliance using the Windows proxy manager and adding the Windows proxy pools, you must add the proxies to the pools. Unless firewalls or security policies prevent it, the easiest way to register a proxy is from the Windows proxy manager.

To add a Windows proxy to a Windows proxy pool with the appliance UI, perform the following steps:

1. On the Windows proxy management page, click the Actions list for the corresponding Windows proxy pool.
2. Select Add Windows Proxy.
   Each proxy pool contains proxies based on:
   - Type of proxies: A proxy pool must contain either Credential proxies or Active Directory proxies. A proxy pool cannot contain proxies of both types.
   - Version of proxies: A proxy pool must contain either version 9.0 and later proxies, or proxies of earlier versions. A proxy pool must not contain proxies of both version 9.0 and later, and proxies of earlier versions.
   Based on the proxy version and version of the Operating System (OS) the proxy runs on, the proxy capability is one of the following:
     - Fully IPv6 capable: Can scan IPv6 addresses and retrieve IPv6 data. Where BMC Atrium Discovery version 9.0 or later proxies are running on Windows 2008 or later.
     - Cannot scan IPv6 addresses: Can retrieve IPv6 data but the Windows version does not support scanning IPv6 addresses. Where BMC Atrium Discovery version 9.0 or later proxies are running on versions of Windows older than Windows 2008.
     - Not IPv6 capable: Cannot scan IPv6 addresses and cannot retrieve IPv6 data. BMC Atrium Discovery proxies from versions older than 9.0.
3. Enter the following information about the new Windows proxy:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Select the Windows proxy pool from the list.</td>
</tr>
</tbody>
</table>
### Field Name Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy Pool</td>
<td>Enter the name that the Windows proxy will be referred to in the user interface. This name must be unique. The system validates this name to be unique.</td>
</tr>
<tr>
<td>Proxy Name</td>
<td>The address of the Windows proxy. This can be specified as one of the following:</td>
</tr>
<tr>
<td>Proxy Address</td>
<td>• Hostname or FQDN</td>
</tr>
<tr>
<td></td>
<td>• IPv4 or IPv6 address</td>
</tr>
<tr>
<td>Port</td>
<td>The port on which to communicate with the Windows proxy.</td>
</tr>
<tr>
<td></td>
<td>• For AD Windows proxy, the default port number is 4321.</td>
</tr>
<tr>
<td></td>
<td>• For credential Windows proxy, the default port number is 4323.</td>
</tr>
<tr>
<td>CA Certificate</td>
<td>The contents of the certificate that was generated automatically by the Windows Proxy Manager. Leave this unset to cause the appliance to automatically retrieve the certificate.</td>
</tr>
<tr>
<td>Enabled</td>
<td>To enable the Windows proxy, select the check box.</td>
</tr>
</tbody>
</table>

⚠️ **When you install a Windows proxy from the Windows proxy manager, the Proxy Address, Port and Certificate fields are pre-populated and Enabled is selected by default.**

4. Click **Apply**.

   The Windows proxy is added to the proxy pool.

### Viewing Windows proxies

The Windows proxy management page displays the list of the available proxy pools and the corresponding proxies in each pool.

Each proxy pool displays the Windows proxies in rows. For discovery tasks, the Windows proxy pool checks for a Windows proxy. If a proxy has not previously been used to discover the endpoint, then for each proxy pool (matching the domain) in the order that the pools appear, a proxy is selected and discovery attempts to contact it:

- if successful then that proxy is used.
- if unsuccessful due to a timeout, then other proxies in the pool are tried.
- if unsuccessful due to an error, then no more proxies in the pool are tried and the remaining pools are tried in the order in which they appear.

The Windows proxies are shown in color-coded boxes where each color represents the level of login success achieved:

- **Green**: 100% success rate.
- **Yellow**: Partial success.
Blue: The credential has never been used. Credential proxies always appear blue. To check their success rate, see the credential list.

Red: 0% success rate.

For each Windows proxy, the following fields are displayed:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the Windows proxy and the IP address and port. A link is also provided showing the last successful use of the credential. This links to the proxy management screen.</td>
</tr>
</tbody>
</table>
| Status     | Displays the status of a proxy in the pool as follows:  
• Windows proxies are reported as unreachable if it has not been pinged successfully by the appliance.  
• Windows proxies are reported active if its version number and release number are equal to or greater than the minimum specified.  
• Windows proxies are reported deactivated if an IPv6 proxy is added to an IPv4 compatible pool.  
For more information on how to enable the Windows proxy, see Editing and managing Windows proxies. |
| Version    | The version and release of the Windows proxy. |
| Notes      | The state of the proxy based on the proxy version, OS version on which the proxy runs, and IPv6 capability. The state is displayed as one of the following:  
• Can get IPv6 data and scan IPv6 addresses: The proxy version is 9.0 or later and runs on Windows 2008 or later OS.  
• Can get IPv6 data but Windows version does not support scanning IPv6 addresses: The proxy version is 9.0 or later and runs on a version of the OS earlier than Windows 2008.  
• IPv6 not supported: The proxy version is earlier than 9.0. |
| Usage      | A summary of the success rate when the Windows proxy has been used, information on failures, and links to DiscoveryAccesses, credential lists and other useful diagnostic pages. |
| Actions    | A list with the following options:  
• Manage — Opens the Manage Windows proxy page. See #Editing and managing Windows proxies.  
• Disable or Enable — Allows you to enable or disable the proxy.  
• Ping — Pings the Windows proxy to ensure that it can be contacted. This is only available if it is enabled. On a successful ping, the information is refreshed. If the ping is unsuccessful, no version number is displayed.  
• Test — Tests connectivity between the Windows proxy, if it is enabled, and a supplied IP address. See #Testing connectivity.  
• Delete — Deletes the entry. |

Testing connectivity

To test the connectivity between a Windows proxy and a supplied IP address:

1. On the Windows proxy management page, select Actions > Test for the Windows proxy.  
2. Enter the IP address to use for the test and click Test.

   The credential tests page is displayed with the test in progress. For more information about the credential tests page, see Testing credentials.

Editing and managing Windows proxies

You can edit and manage the Windows proxies from the Manage Windows proxy page. On the Windows proxy management page, perform one of the following actions:

• Click on the Windows proxy name that you want to edit and manage, or
Click on the **Actions** list of the corresponding Windows proxy and select **Manage**.

The Manage Windows proxy page for the corresponding Windows proxy is displayed. The top section of the Manage Windows proxy page (following the bread crumb navigation trail and preceding the General Details section) displays the following information:

- **Type**: The Windows proxy type (for example, Active Directory)
- **Version**: The version number of the Windows proxy
  
  The version is displayed on a colored banner. The color of the banner indicates whether a proxy is active or not (for example, a Green banner indicates that the proxy is active). For more information about configuring the colored banner, see [Configuring the banner color](#).

The proxy management page contains the following sections which enable you to view the details of and manage the Windows proxy:

- **General Details**
- **Proxy Settings**
- **Proxy Information**
- **Proxy Configuration**

**General Details**

This section list the details used to create the Windows proxy. You can manage the following details about the Windows proxy:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy Pool</td>
<td>The Windows proxy pool for the Windows proxy. If you want to change the proxy pool, select from the corresponding proxy pool list. The list is only populated with pools that contain the same proxy type.</td>
</tr>
<tr>
<td>Proxy Address</td>
<td>The Windows proxy IP address. If you want to change the current IP address, enter the new IP address in the <strong>IP Address</strong> field.</td>
</tr>
<tr>
<td>Port</td>
<td>The port number associated with the Windows proxy. If you want to change the current port number, enter the new port number in the <strong>Port</strong> field.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Displays whether the Windows proxy is enabled or not. You can enable the windows proxy by selecting the check box or disable it by clearing the check box.</td>
</tr>
<tr>
<td>Apply General Details</td>
<td>If you have modified any of the general details and want to apply those, click <strong>Apply General Details</strong>.</td>
</tr>
<tr>
<td>Reset General Details</td>
<td>If you want to reset the current general details, click <strong>Reset General Details</strong>.</td>
</tr>
</tbody>
</table>

**Proxy Settings**

You can manage the following settings of the Windows proxy:
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Log Level          | You can change the log level of the Windows proxy at runtime here in the same way as you can for the service log levels from the main Logging page in BMC Atrium Discovery. For more details, see [Changing Log Levels at Runtime](#). The Log Level list has the following levels:  
  - Debug: Fine-grained informational events that are most useful to debug an application.  
  - Info: Informational messages that highlight the progress of the application at coarse-grained level.  
  - Warning: Potentially harmful situations.  
  - Error: Error events that might still allow the application to continue running.  
  - Critical: Severe error events that might cause the application to abort. |
| Recording Mode     | (8.3 and later proxies only) Displays the recording mode of the Windows proxy. The recording mode can be one of the following:  
  - Record  
  - Playback  
  - None                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Discovery Methods  | (8.3 and later proxies only) Select the discovery methods and their timeouts for the Windows proxy. The available methods are:  
  - WMI  
  - RemQuery                                                                                                                                                                                                                                                                                                                                                                                                                           |
|                    | **Warning:** If you disable RemQuery, some discovery information not available, including network connection details and the contents of files.                                                                                                                                                                                                                                                                                                                                                       |
| Overload Notification | If you select this option, Windows proxy informs the main Discovery service when it is overloaded (that is, the Windows proxy has a large number of requests in progress). In such a situation, Discovery, if possible, will try another Windows proxy from the same Windows proxy pool. However, if you do not select this option, Discovery will wait for all the requests in progress to complete, which can take a long time if the Windows proxy is overloaded. |
| Log Directory Disk Usage | Displays the disk usage of the log directory.                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Record Directory Disk Usage | Displays the disk usage of the record directory.                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| Purge              | Enables you to automatically purge stored data for a specific time period for log files and record data. A selection list enables you to select the purge date. After you have selected the time period to purge data from the Windows proxy, select the Log Files check box, or Record Data check box, or both, and click Apply. The selected data is now set to automatically get deleted on the specified duration.  
  If you want to immediately purge data, select the data (either the Log Files check box, or the Record Data check box, or both) to delete and click Purge Files Now. |
| Apply Proxy Settings | If you have modified any of the proxy settings details and want to apply those, click Apply Proxy Settings.                                                                                                                                                                                                                                                                                                                                                                                     |
| Reset Proxy Settings | If you want to reset the current proxy setting details, click Reset Proxy Settings.                                                                                                                                                                                                                                                                                                                                                                                                              |
Proxy Information

The following information about the proxy is displayed:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>The name and version of the Windows Operating System.</td>
</tr>
<tr>
<td>Processor(s)</td>
<td>Displays the number of processors and processor type.</td>
</tr>
<tr>
<td>Memory</td>
<td>Displays the memory of the appliance and the percentage of memory used.</td>
</tr>
<tr>
<td>Disk Free</td>
<td>The available disk space in MB.</td>
</tr>
<tr>
<td>Notes</td>
<td>Based on the proxy version, OS version on which the proxy runs, and IPv6 capability, one of the following is displayed:</td>
</tr>
<tr>
<td></td>
<td>• Can get IPv6 data and scan IPv6 addresses: The proxy version is 9.0 or later and runs on Windows 2008 or later OS.</td>
</tr>
<tr>
<td></td>
<td>• Can get IPv6 data but Windows version does not support scanning IPv6 addresses: The proxy version is 9.0 or later and runs on a version of the OS earlier than Windows 2008.</td>
</tr>
<tr>
<td></td>
<td>• IPv6 not supported: The proxy version is earlier than 9.0.</td>
</tr>
</tbody>
</table>

IPv6 functionality is not enabled until proxies of versions earlier than 9.0 are upgraded to 9.0 or later, or are removed.

Proxy Configuration

You can view, download, and apply the Windows proxy configuration data from this section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve Current Configuration</td>
<td>• If you want to view the current Windows proxy configuration in XML format, click View winproxy.conf.</td>
</tr>
<tr>
<td></td>
<td>The current Windows proxy configuration in the XML format is displayed.</td>
</tr>
<tr>
<td></td>
<td>• If you want to download the Windows proxy configuration in XML format, click (download).</td>
</tr>
<tr>
<td></td>
<td>You can either choose to open the configuration file with an XML editor or save the file and open it later.</td>
</tr>
<tr>
<td>Apply New Configuration</td>
<td>If you want to apply a new Windows proxy configuration file, perform the following:</td>
</tr>
<tr>
<td></td>
<td>• Click Choose File and navigate to the new Windows proxy configuration XML file.</td>
</tr>
<tr>
<td></td>
<td>• Select the file in the File Upload box and click Open.</td>
</tr>
<tr>
<td></td>
<td>• Click Upload Proxy Configuration.</td>
</tr>
</tbody>
</table>

Additional Windows proxy configuration

The following sections detail additional ways you might configure the Windows proxy outside of the user interface.
To stop a Windows proxy from the appliance command line

The `tw_terminate_winproxy` utility, located in the $TIDEWAY/bin/ directory, sends a request to the Windows proxy to terminate. To use the utility, you must have the `discovery/slave/write` permission. When the utility successfully sends a terminate request to a Windows proxy an audit event is logged. The audit event is called `windows_proxy_process_terminate` and contains the name of the Windows proxy that the terminate request was sent to.

When using the utility, you must always specify a Windows proxy and a user name. Also, if you do not specify a password, you are prompted for one.

### Running proxy as unpriviledged user

If a Windows proxy is not running as either the Local System account or as a member of the Administrators group, `tw_terminate_winproxy` does not stop the Windows proxy. The following error is logged in the Windows proxy log file:

```
ERROR: Failed to terminate slave service: [(5, 'OpenSCManager', 'Access is denied.' )]
```

**Workaround:** Allow the user that the Windows proxy is running to stop the service. This is documented on the Microsoft Support Site.

For more information about the utility and the command line options, see `tw_terminate_winproxy`.

**Windows proxy platform minimum specification**

The following specification provides a guide to the minimum recommended specification for the Windows proxy hardware. This specification has been verified on Microsoft Windows 2003 Service Pack 2:

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>3GHz Intel Pentium® 4 CPU 512k Cache</td>
</tr>
<tr>
<td>Memory</td>
<td>2GB</td>
</tr>
<tr>
<td>Hard disk</td>
<td>60GB</td>
</tr>
</tbody>
</table>

**Windows proxies on Windows XP SP2**

A feature introduced in Windows XP SP2 can cause Windows proxies running on that platform to log the following warning in the Windows system log:

TCP/IP has reached the security limit imposed on the number of concurrent TCP connect attempts.
This means there were more than 10 un-ACKed TCP SYN's in a second (normally from attempting to connect to an invalid address). The patched version of Windows XP interprets this as a potential virus and starts to queue connections. This can cause other network activity on the host to be very slow.

More information about the warning can be found on the Microsoft Support Site.

This feature has also been included in Windows Vista and Service Packs for Windows 2003 Server.

Configuring Windows NIC discovery

Windows NIC discovery is configured in the $TIDEWAY/etc/winproxy.conf file. There are two sections for each card type:

- a regex to identify the card type
- a lookup list for speed, duplex, and negotiation information

The various adapters are configured in the <adapters> section. An annotated excerpt of the winproxy.conf file is shown below:

Adapters section

The adapters section begins with the <adapters> statement

```
<adapters>
```

Search expression

The next statement is a regular expression to search for in the adapter key in the registry. This is specified in the <adapter> statement.

```
<!-- regex to find the adapter type eg. "^Intel(R) PRO/100" -->
<adapter match="^Intel(R) PRO/100">  
```

The search is performed on the hive "HKLM" and the key

```
SYSTEM\CurrentControlSet\Control\Class\ {4D36E972-E325-11CE-BFC1-08002BE10318}
```

Search parameter

You specify the parameter name which in this case is SpeedDuplex.
The names of some network cards might contain special characters (for example, '*'). For these cards, you must specify the actual attribute. For example:

```xml
<adapter-key name="SpeedDuplex" actual="\*SpeedDuplex"/>
```

While defining the network card name, make sure that the adapter-key name matches with the adapter-map name. In this example, it is SpeedDuplex.

With other network cards you might need to search for different parameter names in the registry key, for example:

```xml
<adapter match="^Realtek RTL8139 Family PCI Fast Ethernet NIC">
  <adapter-key name="DuplexMode"/>
</adapter>
```

**Speed Duplex and Negotiation Determination**

The parameter name which you search for returns an integer. The value of the integer represents the particular combination of speed, duplex, and negotiation currently in use on that card. An example, in this case for the SpeedDuplex parameter in the Intel® PRO/100 network card is shown in the Table below.

<table>
<thead>
<tr>
<th>Parameter Value</th>
<th>Speed</th>
<th>Duplex</th>
<th>Negotiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>AUTO</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>HALF</td>
<td>FORCED</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>FULL</td>
<td>FORCED</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>HALF</td>
<td>FORCED</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>FULL</td>
<td>FORCED</td>
</tr>
</tbody>
</table>

An `<adapter-map ParameterName="value">` statement block is entered for each value expected for the parameter name. Contained in this statement are adapter-value statements with name-value pairs for, in this case, speed, duplex, and negotiation. For example:

```xml
<adapter-map SpeedDuplex="1">
  <adapter-value name="speed" value="10"/>
</adapter-map>
```
Example Mapping

An example for the Intel® PRO/100 network card is shown below:

```xml
<adapter-key name="SpeedDuplex" actual="*SpeedDuplex"/>
<adapter-map SpeedDuplex="0">
    <adapter-value name="speed" value=""/>
    <adapter-value name="duplex" value=""/>
    <adapter-value name="negotiation" value="AUTO"/>
</adapter-map>
<adapter-map SpeedDuplex="1">
    <adapter-value name="speed" value="10"/>
    <adapter-value name="duplex" value="HALF"/>
    <adapter-value name="negotiation" value="FORCED"/>
</adapter-map>
<adapter-map SpeedDuplex="2">
    <adapter-value name="speed" value="10"/>
    <adapter-value name="duplex" value="FULL"/>
    <adapter-value name="negotiation" value="FORCED"/>
</adapter-map>
<adapter-map SpeedDuplex="3">
    <adapter-value name="speed" value="100"/>
    <adapter-value name="duplex" value="HALF"/>
    <adapter-value name="negotiation" value="FORCED"/>
</adapter-map>
<adapter-map SpeedDuplex="4">
    <adapter-value name="speed" value="100"/>
    <adapter-value name="duplex" value="FULL"/>
    <adapter-value name="negotiation" value="FORCED"/>
</adapter-map>
<adapter-map SpeedDuplex="6">
    <adapter-value name="speed" value="1000"/>
    <adapter-value name="duplex" value="FULL"/>
    <adapter-value name="negotiation" value="AUTO"/>
</adapter-map>

Finally, the adapter and adapters XML statements are closed.

```xml
</adapter>
</adapters>
Windows proxy permissions
This document describes the user permissions required by the Windows proxy to obtain information from target Windows hosts for each discovery method available to the Windows proxy. The discovery methods are:

- RemQuery
- WMI

RemQuery access and discovery behavior
The RemQuery utility cannot be run as a non-administrator user. You can only create a service as an administrator, which RemQuery needs to do after copying its service to the ADMIN$ share on the remote machine.

If you cannot provide administrator level credentials then you cannot use RemQuery and cannot:

- Get network connection information from basic discovery
- Get files from patterns
- Run commands from patterns

WMI Access and discovery behavior

<table>
<thead>
<tr>
<th>Method</th>
<th>User</th>
<th>Admin user</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getHostInfo</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getDirectoryListing</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getFileSystems</td>
<td>Not available</td>
<td>OK</td>
</tr>
<tr>
<td>getHBAInfo</td>
<td>Not available</td>
<td>OK</td>
</tr>
<tr>
<td>getInterfaceList</td>
<td>No manufacturer</td>
<td>OK</td>
</tr>
<tr>
<td>getPackageList</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getProcessList</td>
<td>No arguments</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>No command path</td>
<td></td>
</tr>
<tr>
<td></td>
<td>No user name</td>
<td></td>
</tr>
<tr>
<td>getRegistryListing</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getRegistryValue</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>getServices</td>
<td>Not available after Windows 2003 SP1</td>
<td>OK</td>
</tr>
</tbody>
</table>

WMI access permission definitions

<table>
<thead>
<tr>
<th>Permission Set</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td></td>
</tr>
</tbody>
</table>
Permission Set | Details
---|---
| DCOM: Remote access enabled
| WMI: Root\CIMV2 namespace: Remote Enable, Account Enable
| WMI: Root\Default namespace: Remote Enable, Account Enable, Execute
| WMI: Root\WMI namespace: Remote Enable, Account Enable
| Admin user | Access as a member of the Administrators group, for example, to scan a Domain Controller, use Domain Controller credentials.

Notes

- `getNetworkConnectionList` is not available using WMI.
- The NIC manufacturer cannot be retrieved by a non-administrator because the Plug and Play Manager is queried and there is no way to grant a non-administrator access to this.
- An error is written in the Windows proxy's log when discovering a Windows 2003 machine as non-administrator. For example:

```
```

This does not lead to any missing information because a different method is then used to retrieve the system's uptime. If the error is a problem, the user can be assigned to the "Performance Monitor Users" group, which allows this WMI query to succeed.

Granting permissions

The following sections list possible ways to grant the various permissions required to a non-administrator user. This should be seen as a guide only.

Setting DCOM permissions

This section describes three methods to grant remote DCOM permission to a user. This is only required for discovery targets running XP SP2 or later or 2003 SP1 or later.

Method 1
Add the user to the Distributed COM Users group. This group was made available in Windows 2003 SP1.

Method 2
Use Group Policy Objects in an Active Directory environment to grant the permission. Using Group Policy Objects is described in [this Microsoft article](#).

Method 3
Use the following steps to configure DCOM permissions on a machine:

1. Select **Start => Run**, enter `dcomcnfg` and click **OK** - this launches the Component Services configuration GUI.
2. Expand **Console Root =\> Component Services =\> Computers =\> My Computer**.
3. Right-click **My Computer** and select **Properties**.
4. Go to the **COM Security** tab.
5. In the Launch and Activation Permissions section, click **Edit Limits...**.
6. Click **Add**.
7. Enter your domain user name or group name in the text entry field and click **Check Names**.
8. Click **OK**.
9. Set the permissions for the user to Allow for Local Launch, Remote Launch, Local Activation, and Remote Activation.
10. Click **OK** to close the permissions dialog, then **OK** again on My Computer Properties. The user should now be able to remotely access DCOM applications including WMI.

### Setting WMI permissions

This method enables you to manually configure WMI permissions on a machine. You cannot configure WMI security with Group Policy Objects.

Use the following steps to configure WMI permissions on a machine:

1. Select **Start =\> Run**, enter `wmimgmt.msc` and click **OK** - this launches the WMI management tool.
2. Right-Click WMI Control (Local) and select **Properties**.
3. Select the **Security** tab in the WMI Control Properties dialog.
4. Expand the Root object.
5. Select the namespace (Root\CIMV2, Root\Default, and Root\WMI in turn) and click **Security**.
6. Click **Advanced**.
7. Click **Add...**
8. Enter your domain user name or group name in the text entry field and click **Check Names**.
9. Click **OK**.
10. Set Apply onto to This namespace only.
11. Select **Allow** for the desired permissions (for example, Remote Enable, Account Enable, and Execute Methods).
12. Click **OK** three times to get back to the WMI Control Properties Security page.

### Setting remote registry permissions

The following article from Microsoft describes how to set remote registry permissions:

http://support.microsoft.com/kb/314837

The user or group must be given read access to the registry key described in this article. Alternatively, the user could be added to the **Backup Operators** group. However this group has a high level of access to the whole system.
Granting user rights

User rights can be granted either from gpedit.msc for local configuration, or using the Group Policy Management Console.

Standalone Windows scanning tool

In certain applications, computer and server assets are located on systems that typically cannot be reached by the BMC Atrium Discovery appliance, either because of network security or because the systems are on an isolated network.

The standalone Windows scanning tool enables you to perform discovery on these standalone systems. It enables you to run BMC Atrium Discovery from a USB drive inserted into the target Windows host and to inventory all assets on that system. The tool gathers data from Windows computers that are not connected to a network, and creates scanner files with the discovered data.

The standalone Windows scanning tool compliments the equivalent UNIX scanner scripts.

This functionality does not permit interactive running of patterns to gather the full depth of data that a Windows proxy provides; however, it enables BMC Atrium Discovery to contain data about hosts that would otherwise be unreachable and to synchronize this information to BMC Atrium CMDB. After the tool is run on the target host, the collected data must be manually uploaded to the appliance.

Warning

The standalone Windows scanner can be used to manually collect a limited set of information from a Windows host. The scanner is designed to be used solely on isolated systems or networks. It is NOT equivalent to a Windows proxy, because it will only collect basic host, process and package information that can be obtained by WMI queries, not additional data such as NIC registry information (for NIC discovery).

Accessing and downloading the tool

The Windows scanning tool is a .zip archive file that can be extracted onto the target system or onto some form of removable storage (such as a USB drive). The .zip archive file contains everything you need to scan the Windows system; no connection to a BMC Atrium Discovery appliance is required.

To access and download the file:

1. From the Discovery tab, click Tools.
2. In the Standalone Windows Scanner area of the Discovery Tools page, click the Download link at the bottom of the page.
3. Extract the .zip file to a directory on a writable a USB flash drive or similar removable media.

**Supported platforms**

The standalone Windows scanning tool is supported on the following platforms:

- Windows XP
- Windows Server 2003
- Windows Server 2003 R2
- Windows Vista
- Windows Server 2008
- Windows Server 2008 R2
- Windows 7
- Windows Server 2012
- Windows Server 2012 R2
- Windows 8
- Windows 8.1

**Using the tool**

⚠️ **Note**

The Windows scanning tool requires the 32 bit Microsoft Visual C++ 2008 runtime DLLs (regardless of whether the Windows operating system being scanned is 32 bit or 64 bit) that are usually installed on the operating system. On a minimal or custom Windows installation, you might need to install the runtime DLLs manually. To do this, use the 2008vcredist_x86.exe installer that is included in the Standalone Windows Scanner .zip file.

The standalone Windows scanning tool runs as a portable console application directly from an external flash drive.

To use the tool:

1. Insert the USB flash drive into the host system and explore the contents.
2. If necessary, change the command options. (For example, it is recommended that you select the target IP address, if known, to avoid the possibility of overlapping IPs).

3. Double-click the `tw_windows_scanner.exe` file.

A command window opens, and the tool begins collecting data. The following example output illustrates what informational messages the tool displays during data collection:

```
U:> tw_windows_scanner
Output file is results\192.168.1.1.scan
Collecting Host Information ... DONE
Collecting Interface Information ... DONE
Collecting Process Information ... DONE
Collecting Service Information ... DONE
Collecting File System Information ... DONE
Collecting Package Information ... DONE
```

### Changing the command options

By default, the scan files are named based on the IP address of the Windows system (the lowest selected by the tool, after it ignores the localhost 127.0.0.1 address). The system will not overwrite existing scan files unless you change the options shown in the following table.

<table>
<thead>
<tr>
<th>Command Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--id XXX</td>
<td>Sets the name of the scan file to <code>results\XXX.scan</code>. Scanner files are automatically named based on the host's first listed IP address. If you do not set the file name using this option, you might inadvertently generate the same result file from two computers.</td>
</tr>
<tr>
<td>--target IPADDR</td>
<td>Sets the IP address to be scanned. For more information, see <a href="#">Changing the IP address to be scanned</a>.</td>
</tr>
<tr>
<td>-x, --overwrite</td>
<td>Specifies to overwrite any existing scan file.</td>
</tr>
<tr>
<td>-q, --quiet</td>
<td>Specifies to not display informational messages</td>
</tr>
<tr>
<td>--wmi-timeout</td>
<td>Specifies to alter the WMI timeout (the default is 2 minutes)</td>
</tr>
</tbody>
</table>

### Changing the IP address to be scanned

In certain cases, the IP address of the Windows system (the lowest selected by the tool, after it ignores the localhost 127.0.0.1 address) might not be correct (for example, a VPN connection). Although these occurrences are uncommon, the tool will display an error message and exit. To help prevent these occurrences, it might be necessary to explicitly specify the IP address you want to use. To do so, run the tool and use the `--target IPADDR` option to set the target system.
It is also possible to scan other systems using this option, as long as your user account has the required privileges (which are typically Administrator privileges). Setting the IP address to a specific target is especially useful for scanning an isolated subnet, because you would only need to insert the tool into one computer to collect data from them all.

**Uploading the data to the appliance**

After you have collected the data, upload it to the appliance. To do this, you use the SCP utility to transfer the files to the appliance as the upload user. For more information on uploading scanner files to an appliance, see [Loading a scanner file onto the appliance](#).

After the data is uploaded, it is processed by BMC Atrium Discovery. For more information about scanner files, see [Standalone UNIX scanning](#).

**Discovering SNMP devices**

This topic provides information and instructions for discovering SNMP devices:

- Configuring SNMP credentials
- Viewing SNMP credentials
- Adding or editing SNMP credentials
-Granting SNMP v3 permissions
-Testing SNMP credentials

**Configuring SNMP credentials**

The discovery system will attempt SNMP queries if remote login attempts have not been successful. However, discovery will attempt SNMP queries, but will only use it if the SNMP port (UDP 161) is open on the target host.

You generally do not need to set the SNMP parameters unless you use a read community other than Public. Different SNMP parameters can be set for different host systems.

Discovery using SNMP is supported for hosts (see the Discovery Platforms page for a complete list) if only an SNMP credential is available for the host's IP address. However, SNMP only provides basic host information, running processes, network connections and installed packages. It does not support interrogating files, HBAs or running OS commands. If a host is discovered using SNMP, Reasoning always checks to see whether a login credential is available for that host as discovered data is richer when a login is achieved. If a login credential is found and used successfully, the host node created using SNMP discovery is updated. In rare cases, duplicate nodes could be created when the host is subsequently discovered using a login credential (for example, this can happen when the IP configuration changes).

**Viewing SNMP credentials**

To view SNMP credentials:

1. From the secondary navigation bar on the **Discovery** tab, click **Credentials**.
2. Click **Devices**.
3. Click **SNMP**.

The SNMP credentials page is displayed and the following information is shown for each credential:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Credential link</strong></td>
<td>This is the first part of the heading link for the credential and displays the range of IP addresses on which this credential is intended to be used. If you click this heading link, the Edit SNMP Credential page is displayed. For more information about this page, see #Setting up SNMP credentials. A link is also provided showing the last successful use of the credential. This links to the Discovery Access for that use.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>A free text description of the SNMP credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td><strong>Usage</strong></td>
<td>A summary of the success rate when the credential has been used, information on failures, and links to DiscoveryAccesses, credential lists and other useful diagnostic pages.</td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td>Additional options used with this SNMP credential (for example, SNMP version). For more information, see the field name-details table for #Setting up SNMP credentials.</td>
</tr>
<tr>
<td><strong>Actions</strong></td>
<td>A drop-down menu with the following options:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Edit</strong> — Select this to edit the credential. The Edit SNMP Credential page is displayed. See #Setting up SNMP credentials for information on the fields and settings available from this page.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Disable</strong> — To disable a credential, select Disable. The credential is marked as disabled in the credential list. When a credential is disabled, this option is replaced with an <strong>Enable</strong> option. To enable the credential, click <strong>Enable</strong>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Delete</strong> — Select this to delete the credential.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Test</strong> — Select this to test the credential. See #Setting up SNMP credentials and #Testing SNMP credentials for more information.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Move to top</strong> — moves the credential to the top of the list.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Move to bottom</strong> — moves the credential to the bottom of the list.</td>
</tr>
</tbody>
</table>

The SNMP credentials are checked in sequence, and the first matching entry is used. After a working SNMP credential is found, further credentials are not checked. To reorder SNMP credentials, drag the credential to the required position in the list.

The SNMP credentials are shown in color-coded boxes. The colors represent the level of login success achieved with that credential:

- Green: 100% success rate.
- Yellow: partial success.
- Blue: the credential has never been used.
- Red: 0% success rate.

**Adding or editing SNMP credentials**

To add/edit SNMP credentials:

1. From the SNMP credentials page, perform one of the following actions:
   a. To add a new credential, click **Add**.
   b. To edit an existing SNMP credential, click **Actions => Edit**.
2. Enter the SNMP credential details as follows:
### Field Name | Details
---|---
Matching criteria | Select "Match All" to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:
- IPv4 address: for example 192.168.1.100.
- IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.
- IPv6 address: for example fda8:7554:2721:a8b3::3.
- IPv6 network prefix: for example fda8:7554:2721:a8b3::/64.

**The following address types cannot be specified**
- IPv6 link local addresses (prefix fe80::/64)
- IPv6 multicast addresses (prefix ff00::/8)
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into *pills*, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field.

There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

**Pills are not currently supported in Opera.**

Enabled | A check box to define whether or not the credential is enabled.

SNMP Version | The SNMP version to use. From the SNMP version list, select one of the following: 1, 2c, or 3. The default is Version 2c.

**SNMP v1/v2c**

Community | Community used for SNMP read access to the defined host(s). For SNMP V1 and V2c credentials only.

SNMP v3 | For SNMP V3 credentials only.

Username | For SNMP V3 credentials only.

Security Level | For SNMP V3 credentials only. Shows the security level selected using the authentication and privacy protocols.

- **noAuthNoPriv**: no authentication and no privacy.
- **authNoPriv**: authentication, no privacy.
- **authPriv**: authentication and privacy.

There is no setting for privacy without authentication.
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Authentication Protocol | The protocol used to encrypt the authentication with the client. For SNMP V3 credentials only. Select one of the following from the drop down list:  
  • None: no encryption used. Operates in the same way as v1 and v2.  
  • MD5: an authentication passphrase is entered and MD5 hashed. The MD5 hashed passphrase is used to access the target system.  
  • SHA: an authentication passphrase is entered and SHA hashed. The SHA hashed passphrase is used to access the target system. |
| Authentication Key | The key (passphrase) which will be used to encrypt the credentials. For SNMP V3 credentials only, and only if you have chosen an authentication protocol. Must be at least 8 characters. |
| Privacy Protocol | The protocol used to encrypt data retrieved from the target. Encrypting the data retrieved from a discovery target causes performance degradation over no encryption. This is for SNMP V3 credentials only, and only if you have chosen an authentication protocol. That is, you cannot have privacy without authentication. Select one of the following from the drop down list:  
  • None: no data encryption is used. Operates in the same way as v1 and v2.  
  • DES: uses a privacy key to encrypt data using the DES algorithm.  
  • AES CFB128: uses a privacy key to encrypt data using the AES algorithm. |
| Private key      | The key (passphrase) which will be used to encrypt the data. For SNMP V3 credentials only, and only if you have chosen a privacy protocol. Must be at least 8 characters. |

3. Click **Apply**.  
The SNMP Credentials page is refreshed to show details of the new credentials.

**Granting SNMP v3 permissions**

When SNMP v3 is used to discover a device that uses different security contexts for different instances of a MIB (in the same way that community string indexing is used for v1 or v2), the SNMP v3 user might not have access to the different security contexts.

If a device is discovered where access to different contexts is required, but access has not been granted to the user, discovery will gather less information and topology discovery might not be complete. A ScriptFailure node will be associated with the DeviceInfo for the DiscoveryAccess, with a message of the type, `Failed to access vlan-1 (AuthorizationError)`, where `vlan-1` is the name of the security context that discovery attempted to access.

To ensure discovery has full access, the user should be granted access to all of the contexts on the network device. For example, to grant access to all contexts to the group `privgroup` on a Cisco device with a recent version of IOS, you can use this configuration command:
You should consult your device documentation or manufacturer for more details.

**Testing SNMP credentials**

When you have added the credentials, you should test them to ensure that they work by performing the following actions:

1. Click **Actions** => **Test** for the SNMP credential.
   A dialog box is displayed with the credential values, and a field in which you enter the IP address against which to test the credential.
2. Enter the target IP address to test.
3. Click **Test**.
   The page is refreshed to show that the test in progress and when complete, the results are shown on the Credential Tests page; this might take a few minutes.

Repeat the preceding steps for all the credentials you want to test.

**Discovering Load Balancers**

BMC Atrium Discovery enables you to discover load balancer instances and their configuration. The resulting inferred model shows how the load balancer components relate to each other. You can discover dedicated hardware load balancers such as Application Delivery Controllers from F5 or Citrix (NetScaler), or the virtualized versions of these devices.

**Network device node (load balancer)**

BMC Atrium Discovery can now discover and model the load balancing configuration for certain NetworkDevice nodes that were previously displayed with a Type of Layer 3 Switch.

Load balancer information is discovered using SNMP. When BMC Atrium Discovery finds a hardware or virtualized Application Delivery Controller (ADC) that has load balancer options enabled, it creates a network device node and triggers the load balancer discovery TPL patterns to query the necessary information about the related load balancing components, including pools, hosts and services. As soon as these elements are discovered, the following nodes (with relationships) are modeled, where applicable:

- Load Balancer Group
- Load Balancer Instance
- Load Balancer Service
- Load Balancer Pool
- Load Balancer Member
The node details page for a NetworkDevice node representing a load balancer has an additional Services section. The discovered objects specific to load balancers are listed in the table below.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instances</td>
<td>A logical load balancer instance representing the load balancing module running on the device.</td>
</tr>
<tr>
<td>Services</td>
<td>A load balancing service that is used for incoming connections to a load balancer.</td>
</tr>
<tr>
<td>Pools</td>
<td>Pools containing members; when pools are not used, a “default” placeholder pool node is created.</td>
</tr>
<tr>
<td>Members</td>
<td>Connection to physical servers linked to the load balancer (in <code>&lt;ip address&gt;:&lt;port&gt;</code> format).</td>
</tr>
<tr>
<td>Hosts</td>
<td>Hosts connected to the load balancer.</td>
</tr>
</tbody>
</table>

An example of the discovered network device with load balancing options enabled:
Discovering Management Controllers

A management controller in BMC Atrium Discovery represents a chassis controller on a blade enclosure, or a lights-out management board on a host. The chassis controller/lights-out management board is currently discovered using SNMP or XML API credentials.

The following links provide information on discovering management controllers:

- Configuring SNMP credentials
- Configuring XML API credentials

When a management controller is discovered, it is modeled as a Management Controller node. The management controller page in the UI displays discovered information.

Configuring XML API credentials

Some management controllers can be discovered using an XML API. The following devices support such discovery:

- Cisco IMC devices
- HP iLO for ProLiant devices

To view XML API credentials

1. From the secondary navigation bar on the Discovery tab, click Credentials.
2. Click Devices.
3. Click XML API.

The XML API Credentials page displays the following information for each credential:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential link</td>
<td>A heading link describing the credential. This consists of the following parts:</td>
</tr>
<tr>
<td></td>
<td>• IP range — the IP address or addresses of the XML API credential is intended to be used.</td>
</tr>
<tr>
<td></td>
<td>• Username — the username used for this credential.</td>
</tr>
</tbody>
</table>

If you click this heading link, the Edit XML API Credential page is displayed. For more information about this page, see to set up XML API credentials below.

<table>
<thead>
<tr>
<th>Description</th>
<th>A free text description of the XML API credential supplied by the user who created the credential.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actions</td>
<td>A drop-down menu with the following options:</td>
</tr>
<tr>
<td></td>
<td>• Edit — Select this to edit the credential. The Edit XML API Credential page is displayed. See to set up XML API credentials for information on the fields and settings available from this page.</td>
</tr>
<tr>
<td></td>
<td>• Disable — To disable a credential, select Disable. The credential is a marked as disabled in the credential list. When a credential is disabled, this option is replaced with an Enable option. To enable the credential, click Enable.</td>
</tr>
<tr>
<td></td>
<td>• Delete — Select this to delete the credential.</td>
</tr>
<tr>
<td></td>
<td>• Test — Select this to test the credential. For more information, see Testing XML API credentials.</td>
</tr>
<tr>
<td></td>
<td>• Move to top — moves the credential to the top of the list.</td>
</tr>
</tbody>
</table>

If you select the options described above, you'll see the corresponding actions.
1. From the XML API credentials page, perform one of the following actions:
   a. To add a new credential, click Add.
   b. To amend an existing credential, click Actions => Edit.

2. Enter the XML API credential details as follows, and click Apply:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching Criteria</td>
<td>Select &quot;Match All&quot; to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:</td>
</tr>
<tr>
<td></td>
<td>• IPv4 address: for example 192.168.1.100.</td>
</tr>
<tr>
<td></td>
<td>• IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.</td>
</tr>
<tr>
<td></td>
<td>• IPv6 network prefix: for example fdab:7554:2721:a8b3::/64.</td>
</tr>
</tbody>
</table>

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

Warning: You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

• To edit a pill, click the pill body and edit the text.
1. To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
2. To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view. Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

3. Pills are not currently supported in Opera.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>A check box enabling you to enable or disable the credential.</td>
</tr>
<tr>
<td>Username</td>
<td>Username used to access the target.</td>
</tr>
<tr>
<td>Password</td>
<td>The corresponding password. In the Edit XML API Credential page, this field is displayed as Set Password. When editing a credential, the password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Enter the password into the password entry field; the password text is not echoed to the screen.</td>
</tr>
<tr>
<td>Timeout</td>
<td>The time (in seconds) in which a response is expected. The default is 180 seconds.</td>
</tr>
<tr>
<td>Description</td>
<td>A free-text description of this credential.</td>
</tr>
<tr>
<td>Protocol</td>
<td>The protocol to use to communicate with the target. Select Cisco CIMC or HP iLO RIBCL. When selecting HP iLO RIBCL you should be aware that after too many login failures, HP iLOs impose a login delay. To avoid this, ensure that the credential is specific to the device so that it is checked first.</td>
</tr>
<tr>
<td>Custom HTTPS Port</td>
<td>To choose a custom HTTPS port, choose from the ports in the list. You must already have configured a custom HTTPS port in the Discovery Configuration window.</td>
</tr>
</tbody>
</table>

The XML API credential page is refreshed to show details of the new credentials.

Testing XML API credentials

When you have added the credentials, you should test them to ensure that they work by performing the following actions:

1. Click Actions => Test for the XML API credential. A dialog box is displayed with the valid IP range, description, username, and options pre-populated.
2. Enter the target IP address to test.
3. Click Test. The page is refreshed to show that the test in progress and when complete, the results are shown on the Device Credentials page; this might take a few minutes.
4. Repeat this for all the credentials you want to test.
Discovering ESX and ESXi hosts

VMware ESX and ESXi are bare-metal/embedded hypervisors, built on a Linux kernel, and run directly on server hardware without requiring an underlying OS. VMware ESXi is a smaller footprint version of VMware ESX. In VMware ESX and ESXi versions before 3.5.0 the Linux environment was available through an ssh connection, with the introduction of version 3.5.0 the ssh capability must be enabled by the user. Discovery using the web services API (introduced with VMware ESX version 3.0.2) is the preferred method of discovering VMware ESX and ESXi.

ESX and ESXi 3.0 and onwards are classified as hypervisors. Previous versions are classified as Linux hosts.

Unpatched VMware vSphere known problems

Unpatched versions of VMware vSphere have known problems when scanned by various tools. We strongly recommend that you apply the appropriate patches to affected systems. There is more information on this issue on the following Configipedia link.

Discovery of VMware ESX and ESXi hosts is performed in the following way:

1. A scan of an IP address detects the following:
   a. Port 902 is open and responds to a vSphere API call with a message from the VMware Authentication Daemon.
   b. Port 443 (HTTPS) is open.
2. If VMware vCenter credentials are defined, then discovery attempts to connect to vCenter on port 443 with a request to discover the target. vCenter then uses vSphere API calls to attempt discovery of the target. If this is successful then the ESX/ESXi host is discovered. There might be multiple vCenter credentials deployed for vCenter servers on the network. These are tried in turn until the host is discovered or the credentials are exhausted.
3. If unsuccessful, discovery attempts to make direct vSphere API calls to the target. This requires vSphere credentials and again, each is tried in turn until the host is discovered or the credentials are exhausted.
4. If vSphere is unsuccessful, and port 22 or an alternative ssh port is configured, an ssh discovery is attempted.

The following screen shows a discovered VMware ESX host.
The following topics provide information and instructions for discovering ESX and ESXi hosts:

- Using vCenter
- Using vSphere API
- VMware ESX and ESXi ssh discovery

Using vCenter

Before you begin

- BMC Atrium Discovery requires valid vCenter credentials to authorize the API calls session (see managing vSphere credentials for more information).
- As discovery uses ports 443 and 902 ports to communicate with the vCenter server, ensure that these ports are open on the vCenter host.

Discovering VMware ESX and ESXi hosts via VMware vCenter Server

VMware vCenter Server provides centralized management of VMware vSphere (ESX and ESXi) virtual machines. BMC Atrium Discovery uses the VMware vSphere API to communicate via a proxy with VMware vCenter to discover VMware ESX and ESXi hosts.

BMC Atrium Discovery can discover ESX and ESXi hosts through the vSphere web services API, or a fallback to an ssh login. However, if a host is being managed by vCenter and it is put into lockdown mode, these discovery techniques are disabled and access is only available through the vCenter server managing it.

To discover an ESX or ESXi host using vCenter, scan the IP address of the target host. Once discovery has determined that the host is an ESX or ESXi host, the discovery request is passed to the vCenter proxy which uses a vCenter credential to pass vSphere API calls through vCenter onto the target hosts.

The API calls described on the vSphere API Support page at Administration > Discovery Platforms > VMware ESX or VMware ESXi are used to discover the system.
Intermittent retrieval of serial number (ServiceTag)

vCenter caches the serial number (ServiceTag) value in memory rather than in its database. That cache expires after some time. So, if you look at the ESX host via the vSphere client, managed object browser or perform a scan while the cached value is held in memory you see the ServiceTag value and BMC Atrium Discovery retrieves it. Once the value has expired the only way to get it back is to restart the ESX host services.

This behavior will not be fixed until the next major vSphere release, vSphere 6.

A discussion on this item is available on the BMC Atrium Discovery Community Forum.

Managing vCenter credentials

This topic covers the following information and procedures for managing vCenter credentials that are used to access the VMware vCenter Server:

- Before you begin
- Adding and editing vCenter credentials
- Testing vCenter credentials
- Viewing vCenter credentials
  - On the Device tab
  - On the Management System tab
  - Credentials color coding
  - Credentials order

⚠️ vCenter credentials are not host credentials

The vCenter credentials that you enter in this page are only used for the vCenter server. Unmanaged, or ESX and ESXi instances that have not been put into lockdown mode can be discovered using VMware vSphere API and require vSphere credentials.

Before you begin
Adding and editing vCenter credentials

To add or edit a vCenter credential:

1. From the vCenter credentials page, perform one of the following actions:
   - To add a new credential, click **Add**.
   - To amend an existing credential, click **Actions > Edit**.

2. Enter the vCenter credential details as follows:

<table>
<thead>
<tr>
<th>Field</th>
<th>Details</th>
</tr>
</thead>
</table>
| Matching Criteria | The IP address or addresses of the vSphere (ESX and ESXi) targets that might be managed by the vCenter server. Select "Match All" to match all endpoints. Deselect it to enter IP address information which determines whether this credential is suitable for a particular endpoint. Enter IP address information in one of the following formats:
  - IPv4 address (for example 192.168.1.100). Labelled v4.
  - IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4.

| The following address type cannot be specified |
| IPv4 multicast addresses (224.0.0.0 to 239.255.255.255) |

As you enter text, the UI divides it into *pills*, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.
- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

<table>
<thead>
<tr>
<th>Field</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>A check box enabling you to enable or disable the credential.</td>
</tr>
<tr>
<td>Name</td>
<td>The name used to identify the connection. It can contain only letters, numbers, and underscores and must begin with a letter or underscore.</td>
</tr>
<tr>
<td>Username</td>
<td>Username used to access the vCenter server. This is not a credential for the target host, but the vCenter server. The permission level required is read-only.</td>
</tr>
<tr>
<td>Password</td>
<td></td>
</tr>
</tbody>
</table>
### Field Name | Details
---|---
| The corresponding password. In the **Edit vCenter Credential** page, this field is displayed as **Set Password**. When editing a credential, the password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Enter the password into the password entry field; the password text is not echoed to the screen.
| vCenter Address | The address of the vCenter server managing the vSphere (ESX and ESXi) targets. This can be specified as one of the following:
- Hostname or FQDN
- IPv4 address
| Port | To choose a custom HTTPS port, choose from the ports in the list. You must already have configured a custom HTTPS port in the **Discovery Configuration** window.
| Timeout | The time (in milliseconds) in which a response is expected. The default is 60 seconds.
| Description | A free-text description of this login credential.

3. Click **Apply**.
   The vCenter credential page is refreshed to show details of the new credentials.

**Testing vCenter credentials**

You might need to test credentials to ensure that BMC Atrium Discovery can use them to successful login into the vCenter.

Testing credentials is available either from the **Device > vCenter** tab, or from **Management Systems > vCenter** tab.

To test vCenter credentials, perform the following actions:

1. (When you use the Management Systems credentials only): Click on the necessary vCenter item and open **Credentials** tab on the vCenter details page that is shown.
2. Click **Actions > Test** for the vCenter credential.
   A dialog box is displayed with the defined IP range, VMware vCenter Server IP address, description, username pre-populated, and a field in which you enter the IP address against which to test the credential.
3. Enter the target vSphere (ESX and ESXi) IP address to test (*not* the vCenter server).
4. Click **Test**.
   The page is refreshed to show that the test in progress and when complete, and the results are shown on the Device Credentials page. This might take few minutes.

**Viewing vCenter credentials**

**On the Device tab**

To view existing vCenter credentials:

1. On the **Discovery** tab, from the secondary navigation bar click **Credentials**.
2. On the **Device** tab, click **vCenter**.

The vCenter credentials page displays the following information for each credential:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential link</td>
<td>A heading link describing the credential. This consists of the following three parts: <strong>Credential name</strong> — the name of the credential. <strong>IP range</strong> — the IP address or addresses of the vSphere (ESX and ESXi) targets that might be managed by the vCenter server. <strong>Username</strong> — the username used to access the vCenter server. This is not a credential for the vSphere (ESX and ESXi) targets, but the vCenter server. If you click this heading link, the Edit vCenter Credential page is displayed. For more information about this page, see #Adding and editing vCenter credentials.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the vCenter server credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td>vCenter address</td>
<td>The IP address or name of the vCenter server.</td>
</tr>
<tr>
<td>Usage</td>
<td>A summary of the success rate when the credential has been used, information on failures, and links to Discovery Accesses, credential lists and other useful diagnostic pages.</td>
</tr>
<tr>
<td>Actions</td>
<td>A drop-down menu with the following options: • <strong>Edit</strong> — Select this to edit the credential. The Edit vCenter Credential page is displayed. See #Adding and editing vCenter credentials for information on the fields and settings available from this page. • <strong>Disable</strong> — To disable a credential, select Disable. The credential is a marked as disabled in the credential list. When a credential is disabled, this option is replaced with an Enable option. To enable the credential, click Enable. • <strong>Delete</strong> — Select this to delete the credential. • <strong>Test</strong> — Select this to test the credential. For more information, see #Testing vCenter credentials. • <strong>Move to top</strong> — moves the credential to the top of the list. • <strong>Move to bottom</strong> — moves the credential to the bottom of the list.</td>
</tr>
</tbody>
</table>

**On the Management System tab**

To view existing vCenter credentials in Management System section:

1. On the **Discovery** tab, from the secondary navigation bar click **Credentials**.
2. On the **Management System** tab, click **vCenter**.

Click to expand more detailed information on vCenter management system credentials...

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the vCenter credential. This field is a link to the Details tab on the vCenter page which contains details about the vCenter credentials, associated patterns, and queries.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of credential. The type is displayed as vCenter.</td>
</tr>
<tr>
<td>Credentials</td>
<td>Displays the number of credentials. This field is a link to the Credentials tab on the vCenter page which contains details about the credential.</td>
</tr>
<tr>
<td>Queries</td>
<td>Displays the number of queries associated with the credential. This field is a link to the Queries tab on the vCenter page which contains details about the queries.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
</tbody>
</table>
| Actions    | A list with the following options:  
- View: Shows the management system details (same as the one described on the step 3 of the procedure that follows).  
- Delete: Not available for the management systems that are in use (dimmed). |

3. Click on the necessary item (for example, vCenter) to view its details. See **Credentials** tab for viewing credentials.

The following table describes the fields for each tab:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
</table>
| Details | This tab has the following sections:  
- Details:  
  - Name: Name of the credential.  
  - Description: A description of what the credential does and how. (For example, *Queries for getting licensing information for ESX/ESXi hosts via vCenter.*)  
  - Type: The type of credential. The type is displayed as vCenter.  
  - Connection setup: Describes whether the connection setup is dynamic or static.  
  - Related Patterns:  
    - Pattern: The list of related patterns. Each listed pattern links to the view pattern page.  
    - Details: The list of queries related to the patterns. Each listed query links to the view query page.  
  - Related Integration Results by Pattern:  
    - Pattern: The pattern that triggered this result. Each listed pattern links to the view pattern page.  
    - State: Describes whether the pattern is active or not.  
    - Results: Displays the integration result and links to the *Integration Result* page. |
| Credentials |  
| Name | Displays the name of the credentials. The credential name links to the credential details tab. |
| TPL | Displays the pattern that triggered this result. |
| Create | To create a new credential, you can select this option. The *Create New Credential* page is displayed. However, the recommended procedure to add a new vCenter credentials is from the **Devices** tab. For more information about adding vCenter credentials from the **Devices** tab, see #Adding and editing vCenter credentials. |
| Actions | This list contains the following options:  
- Edit: You can select this to update existing vCenter credentials. However, the recommended procedure to edit vCenter credentials is from the **Devices** tab. For more information about editing vCenter credentials from the **Devices** tab, see #Adding and editing vCenter credentials.  
- Test: Select this to test the credential. For more information, see #Testing vCenter credentials.  
- Copy: If you want to copy the credential, select this option.  
- Delete: To delete the credential, select this option. |
<p>| Queries |<br />
| Name | Displays the name of the queries. Each query is a link to the view query details page. It also displays whether the query was successfully used or not. |
| TPL | Displays the pattern that triggered this result. |</p>
<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query</td>
<td>Displays the query name.</td>
</tr>
<tr>
<td>Actions</td>
<td>To see the query details, select View.</td>
</tr>
<tr>
<td>Show</td>
<td>From the list, select the number items to view from this page. The available options are 5, 10, 20, 30, and 50.</td>
</tr>
</tbody>
</table>

**Credentials color coding**

The vCenter credentials are shown in color coded boxes. The colors represent the level of login success achieved with that credential:

- Green: 100% success rate.
- Yellow: Partial success.
- Blue: The credential has never been used.
- Red: 0% success rate.

**Credentials order**

The credentials are checked in sequence, and the first matching entry is used. After a working credential is found, no additional credentials are checked. If you want to reorder login credentials, drag the credential to the required position in the list.

**Using vSphere API**

VMware vSphere is a virtual infrastructure management suite. BMC Atrium Discovery uses the vSphere API to communicate directly with VMware ESX and ESXi hosts where no VMware vCenter credentials are available.

### Discovered versions

VMware ESX and ESXi discovery uses version 2.5 of the vSphere API. This supports discovery of the following versions and later:

- ESX 3.5
- ESXi 3.5
- vCenter Server 4.0
- VirtualCenter 2.5

**Before you begin**

- BMC Atrium Discovery requires valid vSphere credentials to authorize the API calls session (see managing vSphere credentials for more information).
• Be aware, that unpatched versions of VMware vSphere have known problems when scanned by various tools. It is strongly recommended that you apply the appropriate patches to affected systems. There is more information about this issue on the following Configipedia link.

• Note, that the Discovery of VMware ESX and ESXi hosts using the VMware vSphere API is only supported over IPv4. For IPv6, BMC Atrium Discovery uses ssh discovery.

Discovering VMware ESX and ESXi systems using vSphere API
To discover VMware ESX and ESXi systems using the VMware vSphere API, BMC Atrium Discovery performs the following actions:

1. Verify that HTTPS port (443) and VMware Authentication Daemon port (902) are open.
2. Establish initial connection with a web services request to port 902 to ensure that the vSphere API is available.
3. Attempt to access the vSphere API using port 443. If the attempt is successful, BMC Atrium Discovery uses the API calls described on the vSphere API Support page at Administration > Discovery Platforms > VMware ESX or VMware ESXi to discover the system.

If the API is not accessible, the discovery falls back to an ssh login method to access the underlying Linux OS that was used in previous BMC Atrium Discovery versions.
Managing vSphere credentials

This topic covers the following information and procedures for managing vSphere credentials that are used to authorize the API calls session:

- **Before you begin**
  - Understanding vSphere credentials: Device versus Management System
  - Required vSphere privileges
  - Pattern-specific privileges (for VMwareVM.VMwareVSphereLicenseDetail pattern)
- Adding and editing vSphere credentials
- Testing vSphere credentials
- Viewing vSphere credentials
  - On the Device tab
  - On the Management System tab
  - Credentials color coding
  - Credentials order

⚠️ vSphere credentials are not host credentials

The vSphere credentials that you enter in this page are only used for the vSphere API. If this is not accessible and ssh discovery is attempted, a separate host login credential for the endpoint is required.

**Before you begin**

Understanding vSphere credentials: Device versus Management System

The vSphere credentials are available for viewing and managing on the following locations:

- **Discovery > Credentials > Device > vSphere** tab.
  This is the default and recommended method for adding or editing vSphere credentials.
- **Discovery > Credentials > Management system > vSphere** tab.
  These credentials are populated when patterns containing vSphere queries are activated. You can manually add vSphere credentials from the Management System tab, however the main designation of these credentials is viewing and testing them.

These are two perspectives for viewing the same credentials. For example, if you edit a vSphere credential on the Devices tab, the same changes are reflected in the credential on the Management System tab.

**Required vSphere privileges**

The minimum privilege required to use the vSphere API for discovery is the **System.View** privilege. This is given by default to all users who can log in, including read only users.

**Pattern-specific privileges (for VMwareVM.VMwareVSphereLicenseDetail pattern)**
Some patterns, such as the **VMwareVM.VMwareVSphereLicenseDetail** pattern require additional privileges to gain complete information. This pattern requires access using a credential with the **Global.Licenses** privilege. Without this, the license key information will either be partially (if discovered via vCenter) or fully (if discovered via vSphere) redacted.

<table>
<thead>
<tr>
<th>Privilege and discovery method</th>
<th>Returned key information</th>
</tr>
</thead>
<tbody>
<tr>
<td>With <strong>Global.Licenses</strong> privilege</td>
<td>30DCK-DUMMY-KEY!!-DUMMY-911F0</td>
</tr>
<tr>
<td>vCenter with <strong>System.View</strong> privilege</td>
<td>30DCK-####-####-####-911F0</td>
</tr>
<tr>
<td>Direct (vSphere) with <strong>System.View</strong> privilege</td>
<td>XXXXX-XXXXX-XXXXX-XXXXX-XXXXX</td>
</tr>
</tbody>
</table>

The same information is returned by the VMware client when accessing the target using the same methods and privileges.

**Adding and editing vSphere credentials**

To add or edit a vSphere credential:

1. From the vSphere credentials page, perform one of the following actions:
   a. To add a new credential, click **Add**.
   b. To amend an existing credential, click **Actions > Edit**.
2. Enter the vSphere credential details as follows:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching Criteria</td>
<td>The IP address or addresses of the vCenter server. Select &quot;Match All&quot; to match all endpoints. Deselect it to enter IP address information which determines whether this credential is suitable for a particular endpoint. Enter IP address information in one of the following formats: • IPv4 address (for example 192.168.1.100). Labelled <strong>v4</strong>. • IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled <strong>v4</strong>.</td>
</tr>
</tbody>
</table>

   **The following address type cannot be specified**
   • IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into **pills**, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu.

**Warning**: You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view. Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.
### Field Name and Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pills are not currently supported in Opera.</td>
<td></td>
</tr>
</tbody>
</table>

#### Enabled
- A check box to define whether or not the credential is enabled.

#### Name
- The name used to identify the connection.
- It can contain only letters, numbers, and underscores and must begin with a letter or underscore.

#### Username
- Username used to access the vSphere API. See #Required vSphere privileges for information on the privileges required.

#### Password
- The corresponding password.
- When editing a credential, the password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Enter the password into the password entry field; the password text is not echoed to the screen.

#### Port
- To choose a custom HTTPS port, choose from the ports in the list. You must already have configured a custom HTTPS port in the Discovery Configuration window.

#### Timeout
- The time (in milliseconds) in which a response is expected. The default is 60 seconds.

#### Description
- A free-text description of this login credential.

3. Click **Apply**.  
   The vSphere credential page is refreshed to show details of the new credentials.

#### Testing vSphere credentials

You might need to test credentials to ensure that BMC Atrium Discovery can use them to successful login into the vSphere.

Testing credentials is available either from the **Discovery > Credentials > Device > vSphere** tab, or from **Discovery > Credentials > Management Systems > vSphere** tab.

To test vSphere credentials, perform the following actions:

1. (When you use the Management Systems credentials only): Click on the necessary vSphere item and open **Credentials** tab on the vSphere details page that is shown.
2. Click **Actions > Test** for the vSphere credential.
   - A dialog box is displayed with the the credential values, and a field in which you enter the IP address against which to test the credential.
3. Enter the target IP address to test.
4. Click **Test**.
   - The page is refreshed to show that the test in progress and when complete, and the results are shown on the Device Credentials page (this can take a few minutes).

#### Viewing vSphere credentials

**On the Device tab**

To view existing vSphere credentials:
1. On the **Discovery** tab, from the secondary navigation bar click **Credentials**.
2. On the **Device** tab, click **vSphere**.

The vSphere credentials page displays the following information for each credential:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Range</td>
<td>This is the first part of the heading link for the credential and displays the range of IP addresses on which this credential is intended to be used. If you click this heading link, the Edit vSphere Credential page is displayed. For more information about this page, see [#Adding and editing vSphere credentials](#Adding and editing vSphere credentials). A link is also provided showing the last successful use of the credential. This links to the Discovery Access for that use.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the vSphere credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td>Usage</td>
<td>A summary of the success rate when the credential has been used, information on failures, and links to DiscoveryAccesses, credential lists and other useful diagnostic pages.</td>
</tr>
</tbody>
</table>
| Actions | A drop-down menu with the following options:  
| Edit | Select this to edit the credential. See [#Adding and editing vSphere credentials](#Adding and editing vSphere credentials) for information on the fields and settings available from this page.  
| Disable | To disable a credential, select Disable. The credential is a marked as disabled in the credential list. When a credential is disabled, this option is replaced with an Enable option. To enable the credential, click Enable.  
| Test | Select this to test the credential. For more information, see [#Testing vSphere credentials](#Testing vSphere credentials).  
| Move to top | moves the credential to the top of the list.  
| Move to bottom | moves the credential to the bottom of the list. |

**On the Management System tab**

To view existing vSphere credentials in Management System section:

1. On the **Discovery** tab, from the secondary navigation bar click **Credentials**.
2. On the **Management System** tab, click **vSphere**.

Click to expand more detailed information on vSphere management system credentials...

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the management system credentials (vSphere). This field is a link to the Details tab on the vSphere page which contains details about the vSphere credentials, associated patterns, and queries.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of management system (vSphere).</td>
</tr>
<tr>
<td>Credentials</td>
<td>Displays the number of credentials that are used when attempting to execute the queries associated with this management system. This field is a link to the Credentials tab on the vSphere page which contains details about the credentials available for this management system.</td>
</tr>
<tr>
<td>Queries</td>
<td>Displays the number of queries associated with this management system. This field is a link to the Queries tab on the vSphere page which contains details about the queries.</td>
</tr>
</tbody>
</table>
| Actions | A list with the following options:  
| View | Shows the management system details (same as the one described on the step 3 of the procedure that follows).  
| Delete | Not available for the management systems that are in use (dimmed). |

3. Click on the necessary item (for example, vSphere) to view its details. See **Credentials** tab for viewing credentials.
The following table describes the fields for each tab:

<table>
<thead>
<tr>
<th>Details</th>
<th>This tab has the following sections:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Details:</td>
</tr>
<tr>
<td></td>
<td>• Name: Name of the credential.</td>
</tr>
<tr>
<td></td>
<td>• Description: A description of what the credential does and how (for example, vSphere queries for getting licensing information from ESX/ESXi hosts).</td>
</tr>
<tr>
<td></td>
<td>• Type: The type of credential.</td>
</tr>
<tr>
<td></td>
<td>The type is displayed as vSphere.</td>
</tr>
<tr>
<td></td>
<td>• Connection setup: Describes whether the connection setup is dynamic or static.</td>
</tr>
<tr>
<td></td>
<td>• Related Patterns:</td>
</tr>
<tr>
<td></td>
<td>• Pattern: The list of related patterns. Each listed pattern links to the view pattern page.</td>
</tr>
<tr>
<td></td>
<td>• Details: The list of queries related to the patterns. Each listed query links to the view query page.</td>
</tr>
<tr>
<td></td>
<td>• Related Integration Results by Pattern:</td>
</tr>
<tr>
<td></td>
<td>• Pattern: The pattern that triggered this result. Each listed pattern links to the view pattern page.</td>
</tr>
<tr>
<td></td>
<td>• State: Describes whether the pattern is active or not.</td>
</tr>
<tr>
<td></td>
<td>• Results: Displays the integration result and links to the Integration Result page.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Credentials</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the credentials. The credential name links to the credential details tab.</td>
</tr>
<tr>
<td>TPL</td>
<td>Displays the pattern that triggered this result.</td>
</tr>
<tr>
<td>Create</td>
<td>To create a new credential, you can select this option. However, the recommended procedure to add a new vSphere credentials is from the Devices tab. For more information about adding vSphere credentials from the Devices tab, see #Adding and editing vSphere credentials.</td>
</tr>
<tr>
<td>Actions</td>
<td>This list contains the following options:</td>
</tr>
<tr>
<td></td>
<td>• Edit: You can select this to update existing vSphere credentials. However, the recommended procedure to edit vSphere credentials is from the Devices tab. For more information about editing vSphere credentials from the Devices tab, see #Adding and editing vSphere credentials.</td>
</tr>
<tr>
<td></td>
<td>• Test: Select this to test the credential. For more information, see #Testing vSphere credentials.</td>
</tr>
<tr>
<td></td>
<td>• Copy: If you want to copy the credential, select this option.</td>
</tr>
<tr>
<td></td>
<td>• Delete: To delete the credential, select this option.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Queries</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the queries. Each query is a link to the view query details page. It also displays whether the query was successfully used or not.</td>
</tr>
<tr>
<td>TPL</td>
<td>Displays the pattern that triggered this result.</td>
</tr>
<tr>
<td>Query</td>
<td>Displays the query name.</td>
</tr>
<tr>
<td>Actions</td>
<td>To see the query details, select View.</td>
</tr>
<tr>
<td>Show</td>
<td>From the list, select the number items to view from this page. The available options are 5, 10, 20, 30, and 50.</td>
</tr>
</tbody>
</table>

**Credentials color coding**

The vSphere credentials are shown in color coded boxes. The colors represent the level of login success achieved with that credential:
- Green: 100% success rate.
- Yellow: Partial success.
- Blue: The credential has never been used.
- Red: 0% success rate.

**Credentials order**

The credentials are checked in sequence, and the first matching entry is used. After a working credential is found, no additional credentials are checked. If you want to reorder login credentials, drag the credential to the required position in the list.

**VMware ESX and ESXi ssh discovery**

VMware ESX and ESXi ssh discovery is used as a fallback when the other methods of discovering VMware ESX and ESXi hosts outlined in Discovering VMware ESX and ESXi hosts fail.

If the vSphere API is not contacted then discovery falls back to an ssh login as used in previous BMC Atrium Discovery versions, and the discovery methods detailed in Administration > Discovery Platforms > VMware ESX or VMware ESXi are used to discover the system. If ssh access has not been enabled, the ESX or ESXi system is not discovered.

VMware ESX and ESXi ssh discovery requires a root user permissions

VMware ESX and ESXi ssh discovery requires a root user permissions. You should login directly as the root user. It is possible to login as a non-root user but such a user cannot close its sessions properly. This results in sessions hanging and inactive sessions building up on the ESXi host.

VMware ESX and ESXi discovery limitation

VMware ESX and ESXi ssh discovery cannot determine any network connection details as there is no equivalent to the `netstat` command.

**Discovering storage**

Storage nodes in BMC Atrium Discovery represent a *storage entity* on the network on which servers can store data. The term storage entity refers to the physical and logical collection of chassis, devices, disks, processors, embedded OS and software that make up storage.

A storage entity is typically a rack mounted device with a large number of disks. Storage entities offer large capacities and redundancy using RAID. Connectivity to the network is usually through a number of ports. Hosts connect to the storage entity through a Storage Area Network (SAN) which
may be Fibre Channel (FC) using fiber optics or IP based. A SAN provides block based storage which appears to the client as a disk. Some other storage entities offer similar capabilities using Network Attached Storage (NAS), which is file based storage and appears to the client as a file server.

**Do I have storage discovery?**

Storage Discovery (BMC Atrium Discovery for Storage) is a separately licensed add-on to BMC Atrium Discovery. It is provided as a TKU file that includes all patterns needed to discover the storage infrastructure. If you have purchased this add-on, you can download the storage TKU from the BMC Electronic Product Distribution (EPD) site and apply it to your BMC Atrium Discovery appliances.

Click here for information on storage discovery on EPD and determining whether it is installed on your appliance...

**Am I entitled to download the storage discovery TKU?**

If you have purchased the BMC Atrium Discovery for Storage add-on, you should be able to see the BMC Atrium Discovery for Storage link on your EPD view. The following screen shows BMC Atrium Discovery for Storage highlighted in yellow. If you have purchased the BMC Atrium Discovery for Storage add-on and you still cannot see it, contact your sales representative.

**Have I installed the storage TKU?**

You also need to upload and activate the BMC Atrium Discovery for Storage TKU. This is accomplished in the normal manner by clicking Upload on the Knowledge Management page (see Uploading knowledge for more information). The following screen shows a Knowledge Management page for an appliance which does not have the BMC Atrium Discovery for Storage TKU applied.

The following screen shows a Knowledge Management page for an appliance which does have the BMC Atrium Discovery for Storage TKU applied. In BMC Atrium Discovery 10.1 patch 1, the installed TKU is shown in the Knowledge Summary with the label Latest Storage, this was not shown in version 10.1 before patch 1. Additionally, the storage patterns are shown in the tree under the heading Storage Systems, which is shown expanded in the screen below.
Even without the BMC Atrium Discovery for Storage TKU, you may see Storage Device nodes and SIs representing storage management systems. These are discovered as part of BMC Atrium Discovery's basic discovery process. The storage patterns in the BMC Atrium Discovery for Storage TKU use the creation or update of Storage Device nodes and storage management system SIs as triggers before performing the deeper discovery of the storage entities.

**Discovering storage**

BMC Atrium Discovery performs discovery of storage entities using the following main approaches:

**Discovery via management software**

Many storage entities are managed using bespoke storage management software. For example, the Hitachi Virtual Storage Platform is managed using Hitachi HiCommand Server. During routine discovery, if storage management software is discovered, an SI representing the software is created. The creation or confirmation of an SI representing storage management software triggers the appropriate storage discovery pattern. The storage discovery pattern interrogates the management software using WBEM credentials and queries to discover any storage entities that it is managing.

When the pattern discovers the managed storage entity it creates a Storage System node to represent it, and other nodes to represent the associated storage components.

**Direct discovery through an embedded provider**

Some storage entities that are managed by storage management software can also be discovered directly, bypassing the management software. This is possible if the storage chassis runs an embedded WBEM/SMI-S provider in the chassis. During routine discovery, if a WBEM/SMI-S provider in the storage chassis is discovered, a Storage Device node representing the chassis is created. The creation or confirmation of the Storage Device node triggers the appropriate storage discovery pattern. The storage discovery pattern interrogates the WBEM/SMI-S provider using WBEM credentials and queries to discover the associated storage components.

When attempting to discover a storage entity via a WBEM/SMI-S provider, you should ensure that WBEM is enabled. In some systems it is disabled by default.

When the pattern discovers the storage entity it creates a Storage System node to represent it, and other nodes to represent the associated storage components.
Direct discovery using SNMP

Some other storage entities, such as the NetApp products, can be discovered directly via SNMP. In this case, when an SNMP discovery of an IP address finds a storage entity, a Storage Device node representing the chassis is created. The creation or confirmation of the Storage Device node triggers the appropriate storage discovery pattern. The storage discovery pattern continues discovery of the storage entity using further SNMP requests to discover the associated storage components.

When the pattern discovers the storage entity it creates a Storage System node to represent it, and other nodes to represent the associated storage components.

What is a Storage Device node?

For storage entities discovered using any of the approaches, the main model consists of a central Storage System node, and associated Storage pools, Storage volumes, Storage processors, Front end FC ports, and Storage connections.

Directly discovered storage entities also have a Storage Device node. The Storage Device node can be regarded in much the same way as the SI representing the storage management software. They are the trigger for the discovery of the storage entity rather than an integral component.

Requirements for full discovery

For a full discovery of a storage system, and the relationships between them, you need the following:

- Storage patterns downloaded from the BMC Electronic Product Distribution (EPD) site.
  Once the patterns are uploaded to the appliance and activated, you can see them in the Knowledge management tree view under Storage Systems.

- To discover storage via management software:
  - Credentials for accessing the storage manager host.
  - WBEM credentials for the storage manager.

- To discover storage directly using the WBEM/SMI-S provider:
  - WBEM credentials for the WBEM/SMI-S provider.

- To discover NetApp storage using SNMP:
  - SNMP credentials valid for the storage device.
  - BMC Atrium Discovery must be on a network that can access the NetApp management ports.
  - Note that IPv6 is not supported in the NetApp MIB, so IPv6 virtual interfaces are not discovered.

- To discover and link storage consumers:
  - Credentials for accessing any host which is a consumer of the storage.
For hosts to be linked to storage, the HBA ID must be discovered.

Discovering the same storage entity using different discovery techniques
It is possible to discover the same storage entity via management software or directly using an embedded WBEM/SMI-S provider. Where this occurs, there may be some differences between the attributes retrieved. For example, it has been observed that the model of some disk drives is missing a suffix when discovered directly, so “STE45085 CLAR450” when scanned via the management system is reported as “STE45085” when scanned directly.

Be aware that if you do discover using both techniques, you will observe some flip flop in the attributes in the discovered storage components.

Relationships between storage and consumers
The relationships between storage components and consumers of storage is independent of the order of discovery. For example, where hosts have already been discovered, they are linked to Storage connections through their World Wide Port Name (WWPN).

Viewing discovered storage systems
For examples of discovered storage systems and the way they are represented in BMC Atrium Discovery, see Viewing a storage system.

Configuring WBEM credentials
WBEM (Web-Based Enterprise Management) is a set of systems management technologies for the management of distributed computer systems. BMC Atrium Discovery uses WBEM to discover storage systems using storage patterns. See discovery.wbemEnumInstances for information on the TPL function used for storage discovery.

To view WBEM credentials

1. From the secondary navigation bar on the Discovery tab, click Credentials.
2. Click Devices.
3. Click WBEM.

The WBEM Credentials page displays the following information for each credential:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential link</td>
<td>A heading link describing the credential. This consists of the following parts:</td>
</tr>
<tr>
<td></td>
<td>• IP range — the IP address or addresses of the WBEM credential is intended to be used.</td>
</tr>
<tr>
<td></td>
<td>• Username — the username used for this credential.</td>
</tr>
<tr>
<td></td>
<td>If you click this heading link, the Edit WBEM Credential page is displayed. For more information about this page, see to setup WBEM credentials below.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the WBEM credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td>Actions</td>
<td>A drop-down menu with the following options:</td>
</tr>
</tbody>
</table>
Field Name | Details
---|---
| **Edit** — Select this to edit the credential. The **Edit WBEM Credential** page is displayed. See to setup WBEM credentials for information on the fields and settings available from this page.
| **Disable** — To disable a credential, select **Disable**. The credential is marked as disabled in the credential list. When a credential is disabled, this option is replaced with an **Enable** option. To enable the credential, click **Enable**.
| **Delete** — Select this to delete the credential.
| **Test** — Select this to test the credential. For more information, see **Testing WBEM credentials**.
| **Move to top** — moves the credential to the top of the list.
| **Move to bottom** — moves the credential to the bottom of the list.

The WBEM credentials are checked in sequence, and the first matching entry is used. After a working WBEM credential is found, further credentials are not checked. To reorder WBEM credentials, drag the credential to the required position in the list.

**To set up WBEM credentials**

**To add or edit a WBEM credential:**

1. From the WBEM credentials page, perform one of the following actions:
   a. To add a new credential, click **Add**.
   b. To amend an existing credential, click **Actions => Edit**.
2. Enter the WBEM credential details as follows, and click **Apply**:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Matching Criteria | Select "Match All" to match all endpoints. Deselect it to enter values that will be used to determine if this credential is suitable for a particular endpoint. They can be one or more of the following, separated by commas:
| IPv4 address: for example 192.168.1.100.
| IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.1.*.
| IPv6 network prefix: for example fda8:7554:2721:a8b3::/64.

- **The following address types cannot be specified**
  - IPv6 link local addresses (prefix fe80::/64)
  - IPv6 multicast addresses (prefix ff00::/8)
  - IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Click here for more information on using the pill UI.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

**Warning:** You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
### Field Name | Details
---|---
• To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.  
• To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.  
Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.  
1 Pills are not currently supported in Opera.
Enabled | A check box enabling you to enable or disable the credential.
Username | Username used to access the WBEM server. This is not a credential for the target host, but the WBEM server. The permission level required is read.
Password | The corresponding password.  
In the Edit WBEM Credential page, this field is displayed as Set Password. When editing a credential, the password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Enter the password into the password entry field; the password text is not echoed to the screen.
Description | A free-text description of this credential.
Timeout | The time (in seconds) in which a response is expected. The default is 180 seconds. WBEM queries may take some time so you may need to increase this timeout.
Protocol | The protocol to use to communicate with the WBEM server. Select HTTP, HTTPS, or both.
Custom WBEM HTTP Port | To choose a custom HTTP port, choose from the ports in the list. You must already have configured a custom WBEM HTTP port in the Discovery Configuration window.
Custom WBEM HTTPS Port | To choose a custom HTTPS port, choose from the ports in the list. You must already have configured a custom WBEM HTTPS port in the Discovery Configuration window.

The WBEM credential page is refreshed to show details of the new credentials.

**Testing WBEM credentials**

When you have added the credentials, you should test them to ensure that they work by performing the following actions:

1. Click **Actions => Test** for the WBEM credential.  
   A dialog box is displayed with the valid IP range, description, username, and options pre-populated.
2. Enter the target IP address to test.
3. Click **Test**.  
   The page is refreshed to show that the test in progress and when complete, the results are shown on the **Device Credentials** page; this might take a few minutes.
4. Repeat this for all the credentials you want to test.
Storage device direct discovery

Storage devices can be discovered directly using WBEM queries. This only performs basic identification of the device rather than the full discovery of the entire storage entity and consuming hosts. Full storage discovery required the Storage TKU to be loaded. At the release of BMC Atrium Discovery 10.1, though subsequent storage TKUs will extend this, the default queries were:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Namespace</th>
<th>WBEM Queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMC</td>
<td>Detects EMC Symmetrix, CLARiiON and VNX devices</td>
<td>root/emc</td>
<td>Enumerate EMC_ArrayChassisSELECT * FROM CIM_SoftwareIdentity WHERE CIM_SoftwareIdentity.InstanceID = 'ManagementServerSoftwareIdentity'</td>
</tr>
<tr>
<td>HP 3PAR</td>
<td>Detects HP 3PAR devices</td>
<td>root/tpd</td>
<td>Enumerate TPD_StorageSystem Enumerate TPD_SystemPackage</td>
</tr>
<tr>
<td>HP P2000</td>
<td>Detects HP P2000 devices</td>
<td>root/hpq</td>
<td>Enumerate HP_TopComputerSystemChassis</td>
</tr>
<tr>
<td>Hitachi</td>
<td>Detects Hitachi AMS, HUS, USP, USP_V and VSP devices</td>
<td>root/smis /current</td>
<td>Enumerate HITACHI_DKCChassis</td>
</tr>
<tr>
<td>IBM DS6000 /DS8000</td>
<td>Detects IBM DS6000 /DS8000 devices</td>
<td>root/ibm</td>
<td>Enumerate IBMTSDS_StorageFacilityChassis Enumerate IBMTSDS_StorageSystem</td>
</tr>
<tr>
<td>IBM SVC</td>
<td>Detects IBM SAN Volume Controller devices</td>
<td>root/ibm</td>
<td>SELECT * FROM IBMTSSVC_Cluster SELECT * FROM IBMTSSVC_ClusterSoftwareIdentity SELECT * FROM IBMTSSVC_ProviderSoftwareIdentity</td>
</tr>
</tbody>
</table>

Discovering database content

This section provides the following information about discovering database credentials:

- Configuring database credentials
- Integration points

Configuring database credentials

To retrieve the Database details, BMC Atrium Discovery needs access to the database as a legitimate user, with privileges sufficient to execute the SQL queries. This topic gives an overview of the database credentials and provides steps for adding new credentials for database discovery.

- Before you begin
- To add new database credentials
- Understanding database credentials
  - Database credentials definition in a pattern
  - Sample Queries
Before you begin

- **Ensure you understand how the database credentials are used in BMC Atrium Discovery**
  See Understanding database credentials for more information.

- **Ensure that necessary patterns are activated**
  Database credentials are bound to the patterns that include sql_discovery definition (for the example, see Database credentials definition in a pattern). There is no one general database pattern that covers any database discovery, so you might want to activate the patterns based on the types of databases you expect to discover.

To add new database credentials

1. Open the Discovery page and click **Credentials**.
2. On the Device Credentials page, open the **Databases** tab.
   You can see the list of database credential groups. Every group corresponds to the particular pattern with an sql_discovery definition and contains queries from that definition and credential for database connection created by BMC Atrium Discovery users.
3. Click **Credentials** in the necessary group.
4. Click **Create**.
5. Enter the following general information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the credential (for example, ExtendedMySQL).</td>
</tr>
<tr>
<td>Enabled</td>
<td>A check box to define whether or not the credential is enabled.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a free text description of the credential.</td>
</tr>
<tr>
<td>Username</td>
<td>The username used to log in to the target database.</td>
</tr>
<tr>
<td>Password</td>
<td>The password corresponding to the username.</td>
</tr>
</tbody>
</table>

6. Select the Database Driver from the list.

   **Note**
   You can manage existing JDBC drivers on the Administration > JDBC Drivers page.

   The page expands to show additional database-specific fields.

7. Specify Database IP Address:
   - If you do not need to limit the credentials use to specific IP addresses (for example, when you do not have a complete list of hosts with databases), select **Match All**.
To use these credentials only for particular IP addresses, deselect **Match All** and enter the endpoint IP address(es) as a comma separated list of values that will be used to determine if this credential is suitable for a particular IP address.

Expand for more information on supported IP address types:

They can be:

- **IPv4 address**: for example 192.168.1.100.
- **IPv4 range**: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.\*.
- **IPv6 address**: for example fda8:7554:2721:a8b3::3.
- **IPv6 network prefix**: for example fda8:7554:2721:a8b3::/64.

⚠️ **The following address types cannot be specified**

- IPv6 link local addresses (prefix fe80::/64)
- IPv6 multicast addresses (prefix ff00::/8)
- IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)

As you enter text, the UI divides it into *pills*, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid.

Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the **Showing n of n** label below the **Range** field. There is no paste option on the context sensitive (right click) menu. **Warning**: You cannot paste a comma-separated list of IP address information into the **Range** field in Firefox. This can crash the browser. You can use a space separated list without any problems. • To edit a pill, click the pill body and edit the text. • To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text. • To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view. Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.

ℹ️ Pills are not currently supported in Opera.

8. Fill in other fields that depend on the database driver:

- For Microsoft SQL Server database:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td></td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>The port number to use to connect to the database. If you leave the <strong>Match Regular Expression</strong> check box clear, any port supplied by the pattern will be used to connect to the database. If the check box is selected, any port supplied which matches the regular expression will be used to connect to the database. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
<tr>
<td>Database Instance Name</td>
<td>The database instance name (Microsoft SQL Server). If you leave the <strong>Match Regular Expression</strong> check box clear, the connection will be made to any database whose name is supplied by the pattern. If the check box is selected, the connection will be made to any database whose name matches the regular expression. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
<tr>
<td>Additional JDBC Parameters</td>
<td>Any additional JDBC parameters to use when making the connection. Select the default value check box and specify a value to be used if none is specified by a pattern. These are specified as key=value pairs in a semicolon separated list.</td>
</tr>
</tbody>
</table>

- **For Oracle database version 8i:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>The port number to use to connect to the database. If you leave the <strong>Match Regular Expression</strong> check box clear, any port supplied by the pattern will be used to connect to the database. If the check box is selected, any port supplied which matches the regular expression will be used to connect to the database. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
<tr>
<td>Service name</td>
<td>The Oracle service name, an alias to an instance, or multiple instances in for example a clustered environment. If you leave the <strong>Match Regular Expression</strong> check box clear, the connection will be made to any database whose service name is supplied by the pattern. If the check box is selected, the connection will be made to any database whose service name matches the regular expression. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
</tbody>
</table>

- **For Oracle database versions from 9i to 11g:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>The port number to use to connect to the database. If you leave the <strong>Match Regular Expression</strong> check box clear, any port supplied by the pattern will be used to connect to the database. If the check box is selected, any port supplied which matches the regular expression will be used to connect to the database. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
<tr>
<td>System ID (SID)</td>
<td>The Oracle System ID, or instance running on the Oracle host. If you leave the <strong>Match Regular Expression</strong> check box clear, the connection will be made to any database whose SID supplied by the pattern. If the check box is selected, the connection will be made to any database whose SID matches the regular expression. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
</tbody>
</table>

- **For Sybase database or PostgreSQL database version 7.2 and higher:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>The database name. If you leave the <strong>Match Regular Expression</strong> check box clear, the connection will be made to any database whose name is supplied by the pattern. If the check box is selected, the connection will be made to any database whose name matches the regular expression. Select the default value check box and specify a value to be used if none is specified by a pattern.</td>
</tr>
</tbody>
</table>
To save the details, click **Apply**.

10. Click the **Details** tab to return to the main Credentials summary page.

### Understanding database credentials

When you activate a Database Pattern (a pattern of sql_discovery type) that aims for deeper database discovery, the corresponding Database Credential Group is created. A Database Credential Group is a container for information used to query databases that includes any number of the following elements:

- **Credential** — Contains the information (for example, database credentials, driver, IP address) to create a connection from BMC Atrium Discovery to the target database.
- **Query** — The SQL query that is passed to the target database to extract the required information. The query is supplied by the pattern.

#### Deep database discovery

Any JDBC Driver that is activated on the **Administration > JDBC Drivers** page can be used to create database connections from patterns and query the database using these connections, being properly authenticated.

The Technology Knowledge Update shipped with BMC Atrium Discovery provides patterns for deep discovery of the following databases:
- IBM DB2 - mainframe only
- IBM Information Management System (IMS) - mainframe only
- Microsoft SQL Server
- MySQL
- Oracle
- Sybase

Database credentials definition in a pattern

This is the definitions section of the MySQL_AB.MySQL_RDBMS_Extended. DatabasesAndTables pattern.

definitions MySQLDetails 1.0""""Queries for MySQL to recover detailed content information"

define showTables  // defines a query called showTables  """"Re

parameters := db_name;end define;end definitions;

Sample Queries

<table>
<thead>
<tr>
<th>Description</th>
<th>Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obtaining a list of Schemas</td>
<td>SELECT username FROM all_users ORDER BY username</td>
</tr>
<tr>
<td>Obtaining a list of Tables within a Schema</td>
<td>SELECT table_name FROM all_tables WHERE owner = %schema%</td>
</tr>
<tr>
<td>Obtaining a list of Tablespaces</td>
<td>SELECT * FROM dba_tablespaces</td>
</tr>
<tr>
<td>Obtaining a list of DataFiles within a Tablespace</td>
<td>SELECT * FROM DBA_DATA_FILES WHERE TABLESPACE_NAME = %tablespace_name%</td>
</tr>
</tbody>
</table>

Integration points

What is an integration point?

An integration point is the container for information used to query centralized databases containing information relating to many hosts in the network. For each host, the integration point can query the database and extract useful information to populate attributes on the host node in the BMC Atrium Discovery model. For example, the name and contact details of people responsible for a particular server is valuable data, but is not generally held on that server. BMC Atrium Discovery enables you to extract such data from a central database by means of integration points.

An integration point contains one or more of each of the following components:

- **Connection** — provides the information (for example, database credentials, driver, IP address) to create a connection from BMC Atrium Discovery to the target database.
• **Query** — the SQL query which is passed to the target database to extract the required information. The query is specified in the pattern used to query the target database.

Integration points and queries are created automatically when a pattern that defines them is uploaded. The two are matched up by the integration point name and the name attribute of a Pattern Definitions block. See the Pattern Language Guide for details of Pattern Definitions. Once an integration point is created, you must populate it with at least one connection and one query before it can be used.

**Viewing integration points**

You view integration points from the Discovery page.

1. From the Discovery page, click the Integration Points tab
2. Click the name of an integration point to view it.

The Details tab on the Integration Point window contains the following sections:

- **Details** — basic description of the integration point
- **Related Patterns** — lists any patterns on the appliance that use this integration point.
- **Results** — lists results for each pattern.

The Connections tab on the Integration Point window shows a list of connections configured for this integration point.

The Queries tab on the Integration Point window shows a list of queries configured for this integration point.

**Row colors in integration points**

The rows have colored backgrounds according to the status of the integration point. The following scheme is used:

- **green** — all successful outcomes each time the integration point is called from a pattern.
- **amber** — some successful outcomes and some failures when the integration point is called from a pattern.
- **red** — all failures each time the integration point is called from a pattern.
- **blue** — the integration point has not yet been called from a pattern.

This color scheme is used for connections and queries too.

**Viewing a connection**

The Connections tab on the Integration Point window shows a list of connections configured for this integration point.

To view a connection, click the connection name from the Connections tab on the Integration Point window.
Viewing a query

The **Queries** tab on the Integration Point window shows a list of queries configured for this integration point.

To view a query, click the query name from the **Query** tab on the Integration Point window.

Creating a connection

An SQL discovery connection provides the information to create a connection from BMC Atrium Discovery to the target database. Once a connection has been made, the database can be queried. A connection can be represented as a URL which should be immediately familiar to a database administrator. This is shown in the preview section of the window. An example TPL call using the connection is also shown.

To Create a connection in an integration point:

1. Click the **Connections** tab on the Integration Point screen.
2. Click **Create** at the top right hand side of the Integration Point screen.

Enter details of the connection.

Which fields are displayed and whether they are required depends on the database driver selected.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the connection. You can only use numbers, letters, or the underscore character (_). This name is used in TPL to call the connection. An example TPL call using the connection name is shown in the Useful Information section.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the connection.</td>
</tr>
<tr>
<td>Username</td>
<td>The database user name which will be used to connect to the database server.</td>
</tr>
<tr>
<td>Password</td>
<td>The password corresponding to the database user name.</td>
</tr>
<tr>
<td>Database Driver</td>
<td>The driver to use to connect to the database server. Select the appropriate driver from the drop down list. BMC Atrium Discovery ships with a limited number of database drivers due to licensing restrictions. See <a href="#">adding new drivers</a> for details of how to add further drivers.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address of the database server.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
</tr>
<tr>
<td>Database IP Address</td>
<td></td>
</tr>
<tr>
<td>Port</td>
<td>The port to use to connect to the database server.</td>
</tr>
<tr>
<td>Database</td>
<td>The database to connect to in the database server.</td>
</tr>
</tbody>
</table>
| Additional Parameters | Any additional parameters that you want to supply to the database driver. These are specified as key=value pairs in a semicolon separated list. For example, you might want to specify that a driver which supports it should connect using SSL and with a non-default timeout:
```
useSSL=true;timeout=60
```
For information about the parameters that can be specified for each driver, you should consult the driver documentation on the internet. Links are provided on the JDBC drivers page. |

3. To save the connection, click **Apply**.
   The Connection window is displayed showing the details of the connection including the example URL and TPL call, and an empty result section. The results section is updated when the connection is used.

**Testing a connection**

Once you have created a connection, you should test it. To test a connection:

1. From the Connection window, click **Test**.
2. The screen is refreshed to show a Test Results section below the Connection Details section.

**Testing a query**

Once you have created a query, you should test it. To test a query:

1. From the Query window, click **Test**.
2. From the Test Query dialog select a connection from the list to use to test the query.
3. Click **Next**.
4. Enter the parameters to be passed with the query.
   The example above shows a query requiring the hostname parameter.
5. To perform the test, click **Test**.
   The screen is refreshed to show a Test Results section below the Query Details section.

**Adding new JDBC drivers**

BMC Atrium Discovery ships with the JTDS driver which can connect to MS SQL Server and Sybase. To connect to other databases, you must download and install JDBC drivers for the databases that you want to use.
If you wish to connect to a database from the list below, then all you need to do is download the driver from the appropriate web site (the URL is available from Uploading new JDBC drivers) and upload it using the Uploading new JDBC drivers page. However, before you can use a new driver that is not in the list, you must provide a properties file for it. The properties file determines how the JDBC URL is constructed. The following properties files are shipped with BMC Atrium Discovery and are located in the $TIDEWAY/data/installed/jdbcdrivers directory:

- Ingres
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL
- Sybase
- JTDS — IPv6 access using JTDS is not currently possible

⚠️ Additional driver support

While it is possible to add new JDBC drivers not specified on the preceding list, these will not have been tested by BMC, and as such no guarantees can be made that they will work with BMC Atrium Discovery as expected. For further information, contact BMC Support.

The $TIDEWAY/data/custom/jdbcdrivers and $TIDEWAY/data/installed/jdbcdrivers directories are both checked for properties files. When you save a driver to the $TIDEWAY/data/custom/jdbcdrivers directory, you can create a new properties file in the same directory, or you can edit one of the supplied properties files in the $TIDEWAY/data/installed/jdbcdrivers directory.

⚠️ Installing driver files manually

Note that if copying files into either of these directories via a command line, it is necessary to restart tideway services to make the change effective.

To write a properties file

1. Create a text file called name.properties where name must be the same as the name of the driver that the properties file relates to. For example, mysql-connector-java-5.0.7.properties is the properties file for mysql-connector-java-5.0.7.jar.
2. The following table describes the entries that you must put in the properties file and also the additional optional entries:
# Mandatory Entries

<table>
<thead>
<tr>
<th>Entry Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>driver.name</td>
<td>The name of the driver. This is the field that is displayed throughout the BMC Atrium Discovery UI to identify this driver.</td>
</tr>
<tr>
<td>driver.class</td>
<td>The main class that will be loaded from the relevant jar file in custom/jdbcdrivers. The user will need to get this from the documentation provided by the database/JDBC vendor. For example, driver.class=com.mysql.jdbc.Driver</td>
</tr>
<tr>
<td>driver.url</td>
<td>The format of the connection URL that this driver uses. There is no standard URL for the JDBC URL so every driver has its own format. BMC Atrium Discovery passes the assembled URL to each registered driver and the driver responds whether it recognises the URL or not. The URL is made up of several parts and these are described in JDBC drivers.</td>
</tr>
</tbody>
</table>

# Optional Entries

<table>
<thead>
<tr>
<th>Entry Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>driver.default</td>
<td>Specifies that this driver is the default driver and will be displayed first in lists when creating objects that use JDBC drivers, for example, SQLIntegrationConnections. If no default is specified across all the drivers then the order in which they are displayed is random. If more than one driver is specified as the default then the last driver to be loaded will be marked as the default.</td>
</tr>
<tr>
<td>driver.exceptionProcessor</td>
<td>Specifies a Java class that can format SQL exceptions (java.sql.SQLException) that bubble up from the JDBC driver. This class must implement the ExceptionProcessorIF interface. When an SQL exception occurs, the specified class will be called and then the result of that call will be published to the relevant source, which might be a log file or to the UI. For example, driver.exceptionProcessor=com.tideway.integrations.service.servants.MySqlExceptionProcessor</td>
</tr>
<tr>
<td>translation.variablename</td>
<td>Translates a variable that is specified in the JDBC URL to a user visible label. See the table below for a list of variable names. (for example, translation.extra_parameters=Additional Parameters)</td>
</tr>
<tr>
<td>validationregex.variablename</td>
<td>Used only with Integration Points and Software Credential Groups. Validates a given variable specified in the JDBC URL. When the user tries to save a connection, each of the properties is checked to see if it has an associated validation regex. If so, and if the Match on this check box is NOT selected, or the connection is static, then the value entered by the user is matched against the value specified here. If they do not match, the user is prompted to re-enter the value. Endpoint is always validated, for static credentials, to ensure that it is a valid IPv4 address.</td>
</tr>
</tbody>
</table>

The following table shows the parameters used by each driver:

<table>
<thead>
<tr>
<th>Field</th>
<th>MySQL</th>
<th>Postgres</th>
<th>Oracle (service)</th>
<th>Oracle (SID)</th>
<th>Ingres</th>
<th>Sybase</th>
<th>MS SQL Server</th>
<th>JTDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>server_type</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
</tr>
<tr>
<td>endpoint</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
<td>Req</td>
</tr>
<tr>
<td>instance_name</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
When you have added a new driver and written a properties file for it, you must restart the Tideway services. To do this, enter the following command:

\[ $ sudo /sbin/service tideway restart \]

### Multiple drivers using the same JAR (Oracle)

There are two methods of connecting to Oracle, OracleService and OracleSID. These methods use the same driver JAR file, but need two properties files to build the different connection URLs used. Rather than copying and renaming the JAR file, a better method is to use symbolic links.

The driver file is called \texttt{ojdbc14.jar} for Oracle to 10g. For Oracle 11g the driver file is called \texttt{ojdbc15.jar} which requires JRE 1.5 or later.

\begin{verbatim}
 cd ~/data/installed/jdbcdrivers
 ln -s ojdbc14.jar oracleService.jar
 ln -s ojdbc14.jar oracleSID.jar
\end{verbatim}

This creates a symbolic link called \texttt{oracleService.jar} for the \texttt{oracleService.properties} properties file and one called \texttt{oracleSID.jar} for the \texttt{oracleSID.properties} properties file.

### Uploading new JDBC drivers

A JDBC driver is shipped in the form of a single file with an extension of .\texttt{jar} (Java ARchive). The JAR file contains a compiled java implementation of the client code to remotely talk to the database. Database vendors usually make their JDBC drivers available for free download from their websites.

Due to licensing issues we cannot ship the drivers for all of the databases that we support. They are available from the vendors’ websites for download. We supply the URLs to access these jar files on the JDBC Drivers page. To access this, in the Administration page, Appliance section, click JDBC Drivers.

After you have downloaded the JDBC driver for the database you wish to communicate with, upload it to the appliance using the correct Upload link in the table. The file will be uploaded and verified. If the required Java class is not found in the jar, or if the jar is corrupt (or not a jar at all) then an appropriate message will be displayed in the Status column. Simply re-upload the correct driver to correct the issue should one arise.
tideway service restart required

You must restart the tideway services before using the newly uploaded JDBC driver.

The available statuses for the drivers are:

- No JAR Uploaded
- Jar Uploaded (Deactivated)
- Activated and in use. This means the driver is being used by an SQL credential.
- Activated but not in use
- Error.

BMC Atrium Discovery is shipped with a default set of properties files which define the databases that you can connect to. If you want to connect to a different database, you must write a properties file and download an appropriate driver. This is described in Adding new JDBC drivers.

JDBC drivers

The following table provides an example JDBC URL for database targets. It also provides documentation and download URLs where available. When selecting the appropriate driver, note that BMC Atrium Discovery uses JDK 1.8. Consult the database vendor's documentation for further information.

<table>
<thead>
<tr>
<th>Database</th>
<th>JDBC URL</th>
<th>Internet site</th>
</tr>
</thead>
<tbody>
<tr>
<td>Postgres</td>
<td>jdbc:postgresql://host:[port]/database [?propertyName1=propertyValue1&amp;propertyName2=propertyValue2]</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>JDBC URL</td>
<td>Internet site</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>---------------</td>
</tr>
</tbody>
</table>
| **Oracle** | For Oracle there are two possible connection styles: **Connection using service**
```
jdbc:oracle:<driver type>:[username/password]@//host[:port]/service
```
The following example URL connects, using a thin driver, to an Oracle server running on a host on IP address 192.168.0.100, port 1521, and service ADDM_DB.
jdbc:oracle:thin:@//192.168.0.100:1521/ADDM_DB | **Connection using SID**
```
jdbc:oracle:<driver type>:@host[:port]:SID
```
The following example URL connects, using a thin driver, to an Oracle server running on a host on IP address 192.168.0.100, port 1521, and SID ADDM100.
| **Ingres** | 
```
jdbc:ingres://host:[port]/database
[property=value][;property=value]
```
The following example URL connects to an Ingres server running on a host on IP address 192.168.0.100, port mnemonic II7, database name ADDMdatabase, user fred, and password password.
| **Sybase** | 
```
jdbc:sybase:Tds:host[:port]/[databasesname]
```
<p>|</p>
<table>
<thead>
<tr>
<th>Database</th>
<th>JDBC URL</th>
<th>Internet site</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The following example URL connects to a Sybase server running on a host on IP address 192.168.0.100, port 6689, and database name ADDM_DB. jdbc:sybase:Tds:192.168.0.100:6689/ADDM_DB</td>
<td>Documentation: <a href="http://www.sybase.com/detail?id=1009876#sec2q2">http://www.sybase.com/detail?id=1009876#sec2q2</a></td>
</tr>
<tr>
<td></td>
<td><strong>MS SQL</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>JTDS</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>The following example URL connects using JTDS to an MS SQL server server running on a host on IP address 192.168.0.100, port 1433, database name ADDM_IMPORT, instance TDA, user fred, and password password. jdbc:jtds:sqlserver://192.168.0.100:1433/ADDM_IMPORT;instance=TDA;user=fred;password=password</td>
<td>Documentation: <a href="http://jtds.sourceforge.net/faq.html#urlFormat">http://jtds.sourceforge.net/faq.html#urlFormat</a> When using a domain credential (Windows Authentication) of the form DOMAINNAME\username enter the username in the URL described, and the domain information in the Additional JDBC parameters dialog box in the following form: domain=&quot;DOMAINNAME&quot;. Also, if the domain controller requires NTLM v2 add the parameter: useNTLMv2=true. The following example URL connects using JTDS to an MS SQL server server running on a host on IP address 192.168.0.100, port 1433, database name ADDM_IMPORT, instance TDA, Windows user fred in the domain DOM1, and password password. jdbc:jtds:sqlserver://192.168.0.100:1433/ADDM_IMPORT;instance=TDA;domain=DOM1;useNTLMv2=true;user=fred;password=password</td>
</tr>
</tbody>
</table>
Connecting to DB2 databases

You can use BMC Atrium Discovery to connect to IBM DB2 databases, though this does require some additional configuration over that required for the other supported databases. To do this, you need to perform the following steps:

To configure a connection to a DB2 database

To configure a connection to a DB2 database, you must create a custom properties file and copy it and DB2 JAR and license files to the appliance. Creating a properties file is described in Adding new JDBC drivers. An example properties file is shown below.

```
driver.class=com.ibm.db2.jcc.DB2Driver
driver.name=IBM DB2 driver
driver.url=jdbc:db2://%endpoint%:%port%/%sid%
driver.serverName=%endpoint%
translation.sid=System ID (SID)
translation.endpoint=Database IP Address
validationregex.port=[0-9]+
download.url=http://www.ibm.com
download.label=Driver download page at ibm.com
download.notes=An IBM Account is required.
driver.enabled=true
```

1. Create the properties file and save it as `db2jcc.properties`
2. Copy the `db2jcc.properties` file to the `/usr/tideway/data/custom/jdbcdrivers` directory on the appliance.
3. Copy the DB2 JAR file `db2jcc.jar` to the `/usr/tideway/data/custom/jdbcdrivers` directory.
4. Copy the DB2 license JAR file, for example, `db2jcc_license_cu.jar` to the `/usr/tideway/java/integrations/lib` directory.
5. Restart the tideway service.

Supporting new databases types

Once the JDBC driver is added, you must create a custom pattern with a query definition block. The activation of this pattern allows the creation of credentials for the new database group specified in the custom pattern.

Installing DB2 license jar onto a cluster
Files placed in /usr/tideway/java/integrations/lib are not automatically distributed around a cluster. If using a cluster, the DB2 license jars need to be manually copied to this directory on each cluster member.

**Discovering WebLogic**

Oracle WebLogic server is an enterprise-class J2EE application server. Oracle WebLogic server is part of the Oracle WebLogic platform and supports Oracle, IBM DB2, Microsoft SQL Server, MySQL, and other JDBC-compliant databases.

**Discovering WebLogic**

BMC Atrium Discovery uses the BEA.WebLogicApplicationServer pattern to discover a WebLogic application server instance and its version, to identify the JMX port, and most importantly for extended discovery, to determine whether JMX access is enabled.

Extended WebLogic discovery attempts to discover detailed information related to the WebLogic application server such as the J2EE domain, the J2EE applications running on the WebLogic application server, and the JDBC resources that the WebLogic application server is using. To do this, an additional pattern, WebLogic.ExtendedDiscovery, initially determines whether JMX access is enabled, and the JMX port has been identified. It then attempts to determine whether the WebLogic application server version is supported (see Supported product versions). If the WebLogic server version cannot be determined using a JMX query, then pattern execution ends.

If the JMX port has not been identified, the extended WebLogic discovery uses port 7001 by default. The WebLogic.ExtendedDiscovery pattern then queries the WebLogic Administration Server’s JMX monitoring agent for details about the J2EE applications, application servers, databases, database servers, mail servers, web servers, J2EE domain, and J2EE clusters. Information returned is stored in the J2EEApplication Component, J2EEDomain Collection, JDBCResource Detail, or JavaMailResource Detail nodes.

Creation of a JDBCResource Detail node triggers the CreateJDBCToDatabaseSI pattern that searches the BMC Atrium Discovery model for a software instance (SI) representing that database. If the database SI is found, the pattern creates relationships between the WebLogic Application Server SI and the nodes representing the database (see Database nodes and relationships). If the host that the database runs on has not been scanned, no further work is undertaken. The WebLogic.ExtendedDiscovery.DiscoverWebLogic pattern is fully described in Configipedia.

Creation of a J2EEApplication Component node does not trigger any further operations.

If extended discovery fails, it falls back to using a host login and extracts information from the WebLogic configuration files to create and populate a WebLogic Application Server SI and its attributes.
Requirements for a full discovery

For a full discovery of WebLogic you need the following credentials:

- WebLogic host login credentials
- WebLogic application server login credentials (JMX) — see Configuring extended WebLogic discovery

Database nodes and relationships

The relationships are created based on the way that the database type is represented. (For example, an Oracle database is represented as an SI whereas a MySQL database server is represented as an SI and the individual databases by Detail nodes with contained by relationships to the database server SI).

The use of detail nodes is not intended to be permanent.

Supported product versions

The following WebLogic versions are supported:

- Oracle WebLogic Server 8.1
- Oracle WebLogic Server 9.x
- Oracle WebLogic Server 10.3 to 11g

Extended WebLogic discovery results

The following screen shows a successfully scanned WebLogic instance that is using an Oracle database server as its datasource. If a Microsoft SQL Server database is used, the Database Elements This Depends On row is displayed to indicate the actual database, because in this case the database server is represented by an SI. See Database nodes and relationships for more information.

The attributes section of the WebLogic SI contains sections for Components, Details, and Collections.

- **Components** are deployed EAR and WAR modules on WebLogic
• **Details** are resources on WebLogic. These might be application specific or globally defined. They can be one of the following:
  - Java Mail Resources
  - JDBC Resources
• **Collections** are J2EE Domains on WebLogic

**Configuring extended WebLogic discovery**

The extended WebLogic discovery is enabled by the activation of the WebLogic [ExtendedDiscovery](#) pattern. This pattern is activated by default in a new installation of BMC Atrium Discovery, but in an upgraded appliance, it must be activated manually.

To use extended WebLogic discovery, you must configure the credentials for the WebLogic application server in the Credentials tab.

**Middleware credential always required**

You must set up a middleware credential to perform extended discovery of WebLogic even if the target system requires no authentication. In this case, any values can be entered in the username and password fields.

**To configure extended WebLogic discovery**

1. Ensure that the WebLogic.ExtendedDiscovery pattern is activated.
2. From the Discovery home page, click **Credentials**.
3. Click the **Middleware** tab.
4. Click the **Oracle WebLogic** credential group heading.
5. Click the **Credentials** tab.
6. From the Credentials tab, select **Create**.

Enter the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the Credential (For example, ExtendedWebLogic).</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a free text description of the Credential.</td>
</tr>
<tr>
<td>Username</td>
<td>The logon ID with which to connect to the WebLogic application server. You must specify a logon ID that has a security role of Monitor, Operator, Deployer, or Admin.</td>
</tr>
<tr>
<td>Password</td>
<td>The password corresponding to the username above.</td>
</tr>
</tbody>
</table>
| IP Address | The IP address of the WebLogic host. This option can be one of the following:  
  • an IP address (10.10.10.3)  
  • a range specification (10.10.10.* or 10.10.1-5.* or 10.10.0/24)  
  • a regular expression matching an IP address (.* or 10.10.10.(23|25)) |

7. To save the details, click **Apply**.

The new credential is displayed.
What is a Middleware Credential Group?

A Middleware Credential Group is a container for information used to query middleware such as web and application servers and similar tools that support application development and delivery. They contain one or more credentials and one or more queries.

- **Credential** — contains the login credentials and IP address required to create a connection from BMC Atrium Discovery to the middleware target.
- **Query** — the query which is passed to the Expert Discovery Module (EDM). Only one query is supported by EDMs, that query is `scan` which triggers a discovery scan of the target.

A Middleware Credential Group is created by the activation of a Middleware Pattern. A Middleware Pattern has its type defined as one of the following:

- `tomcat_discovery`
- `weblogic_discovery`

Configuring a Middleware Credential Group

This procedure shows you how to configure a Middleware Credential Group. The only items requiring configuration through the UI for a Middleware Credential Group are the credentials (username, password, connection information). The query comes from the middleware pattern.

The example used is for Tomcat and uses the `Tomcat.ExtendedDiscovery.DiscoverTomcat` pattern.

There is more to configuring the discovery of middleware than configuring a Middleware Credential Group. For a full description of this, see the Discovering Tomcat section.

This is the definitions section of the `Tomcat.ExtendedDiscovery.DiscoverTomcat` pattern. Additional comments explain various definitions.

```plaintext
definitions EDMRequest 1.0"""Queries to recover detailed Tomcat information"

type := tomcat_discovery; // tomcat_discovery places the credentials
```

Discovering Tomcat

Apache Tomcat is an Application Server, otherwise known as a Web Container, developed by the Apache Software Foundation (ASF).

Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server, in addition it adds tools for configuration and management, which can also be configured by editing configuration files that are normally XML-formatted, and includes its own internal HTTP server.
Discovering Tomcat

BMC Atrium Discovery uses the ApacheFoundation.Tomcat.Tomcat pattern to discover the Tomcat instance, identify the home directory (catalina.home), identify the base directory (catalina.base), and determine the Tomcat version. The pattern extracts information from the Tomcat configuration files to create and populate a Tomcat SI and its attributes.

Extended Tomcat discovery is enabled by the activation of the Tomcat.ExtendedDiscovery.DiscoverTomcat pattern. This is activated by default in a new installation of BMC Atrium Discovery. In an upgraded appliance, it must be activated manually. The Tomcat.ExtendedDiscovery.DiscoverTomcat pattern triggers on the creation or update of a Tomcat SI. The Tomcat.ExtendedDiscovery.DiscoverTomcat pattern is fully described in Configipedia.

Creation of a "JDBC Resource" Detail node triggers the CreateJDBCToDatabaseSI pattern which searches the BMC Atrium Discovery model for an SI representing that database. If the database SI is found, the pattern creates relationships between the Tomcat Application Server SI and the nodes representing the database (see below). If the host that the database runs on has not been scanned, no further action is taken.

Creation of a "J2EEApplication" Software Component node does not trigger any additional operations.

See the Tomcat documentation for additional information about Tomcat datasources.

Requirements for a full discovery

For a full discovery of Tomcat, the database or databases that it is using, and the relationships between them, you need the following credentials:

- Tomcat host login credentials
- Database host login credentials
- Database credentials

Tomcat Discovery results

The following screen shows a successfully scanned Apache Tomcat instance.
The attributes section of the Tomcat SI contains sections for Components, and Details.

- **Components** are the Software Component nodes representing J2EE applications on Tomcat.
- **Details** are detail nodes representing resources on Tomcat. These might be application specific or globally defined. They can be one of the following:
  - Custom Resources: a custom resource can be any kind of JavaBean declared as a resource.
  - User Database Resource: the default Tomcat user management database.
  - Java Mail Resources
  - JDBC Resources

**Improvements over extended discovery using JMX**

Previously, extended Tomcat discovery required JMX configuration to ensure that JMX could permit local and remote monitoring. Additionally, discovery using JMX only permitted discovery of globally defined resources. Discovery using configuration files enables you to also discover application specific resources.

**Discovering mainframe computers**

Mainframe computers are powerful computers that tend to be used by large organizations for critical bulk data processing. They are characterized by high reliability and security, strict backward compatibility with older software, and high utilization rates and very large data throughput.

See **Viewing a mainframe** for a description of how a mainframe is represented in BMC Atrium Discovery.

**Configuring mainframe discovery**

BMC Atrium Discovery performs mainframe discovery as part of a normal discovery scan. When discovery heuristics indicate a mainframe computer is on a particular IP address, then the methods unique to mainframe discovery are initiated. Mainframe discovery methods use an agent (the z/OS Discovery Agent), which is installed on the mainframe to collect information from the BMC MainView instance on each logical partition (LPAR).

**BMC Atrium Discovery for z/OS agent**

BMC Atrium Discovery cannot natively discover z/OS systems and requires **BMC Atrium Discovery for z/OS** to be installed on the z/OS systems. This is a separately licensed product. Unless the support ID you use has been registered as licensed for this additional product then you might be unable to see the references linked to on this page. Contact your BMC sales representatives if you need to discuss this further.
Discovering all mainframes and LPARs

To discover all mainframes and LPARs, a BMC Discovery for z/OS agent must be installed on every LPAR. Where LPARs are connected through CAS to CAS communication (known as a CASPlex), then at least one MainView Explorer (host server) must be running on each CASPlex. You must then scan the IP addresses for each of the MainView Explorer instances.

Where LPARs are not connected through the CAS then a MainView Explorer (host server) instance must be running on each LPAR.

For information on installing and configuring the BMC Discovery for z/OS Agent, see the *BMC Discovery for z/OS Agent Installation Guide* available from the BMC support site.

Mainframe discovery limitations

Mainframe discovery only supported over IPv4

Mainframe discovery is only supported over IPv4 as BMC MainView, on which the agent relies, is not supported on IPv6.

Required z/OS Discovery Agent version

Supported versions of the z/OS Discovery Agent are 1.6 and 1.7. Version 1.7 enables discovery of peer-to-peer and client-server relationships between Software Instance nodes. To enable discovery by BMC Atrium Discovery, the z/OS Discovery Agent should be patched according to Technical Bulletin 172149. This Technical Bulletin is available from the BMC support site: Navigate to the page **BMC Discovery for z/OS** and choose the document "Describes BMC Atrium Discovery for z/OS License Add-On 1.6.00 requirements for BMC Atrium Discovery and Dependency Mapping 8.2.0".

LPAR naming and SMF ID

BMC Atrium Discovery sets the names of LPARs (the CAS system names) on the `__discovery_context` attribute of the DiscoveredMainframe node. Sometimes, these names are not the same as the SMF ID of the LPAR that the agent returns to BMC Atrium Discovery. In such cases, there might be partial or no results for discovery methods other than `getMainframeInfo`. The following Program Temporary Fixes (PTFs) address this issue:

- BMC Discovery for z/OS Agent 1.6: PTFs BPA1472 and BPA1473.
- BMC Discovery for z/OS Agent 1.7: PTFs BPA1479 and BPA1480.

Multiple mainframe discovery

Multiple Mainframe nodes can be created in the inferred model. A scan of a BMC Discovery for z/OS agent results in only one inferred mainframe (the one on which the agent is running). The following illustration shows an example visualization.
This screen illustrates a visualization of multiple Mainframe nodes created from the inferred model.

A separate agent must be configured for each additional mainframe to be discovered, even if agents for the mainframes are communicating with each other and each agent is able to gather details of more than one mainframe.

The ability to infer multiple Mainframe nodes depends on the following PTF being applied on the BMC Discovery for z/OS agent:

- BPA1401

If the PTF is not applied, BMC Atrium Discovery will not be able to infer multiple mainframes (the orphan nodes previously described, however, will no longer be created), behavior that existed in version 8.2.01 of the product.

Optional Additional Discovery Tools

Some of the detail data that the z/OS Discovery Agent collects is dependent on additional MainView products being installed. The following table shows the MainView products used to obtain additional detail information. See the BMC Discovery for Z/OS documentation for further information.

<table>
<thead>
<tr>
<th>Method</th>
<th>Script Views</th>
</tr>
</thead>
<tbody>
<tr>
<td>getMainframeInfo</td>
<td>View Z</td>
</tr>
<tr>
<td>getMFPart</td>
<td>View Z</td>
</tr>
<tr>
<td>getApplication</td>
<td>Views 49Z, 4HZ and 4WZ (requested as View 49HWZ) — Requires MainView for WebSphere, Application Server, and MainView Transaction Analyzer products to be installed.</td>
</tr>
<tr>
<td>getCouplingFacility</td>
<td>View Z</td>
</tr>
<tr>
<td>getDatabase</td>
<td>Views 1IZ and 1XZ (requested as View 1IXZ) — Requires MainView for DB2 and MainView for IMS products to be installed.</td>
</tr>
<tr>
<td>getDatabaseDetail</td>
<td>Views 1IJZ, 1ILZ and 1IKZ (requested as View 1JKLZ) — Requires MainView for DB2 product to be installed.</td>
</tr>
</tbody>
</table>
1. **Method**
   - getDependency
   - getDiskDrive
   - getMQDetail
   - getSoftware
   - getStorageSubsystem
   - getSysplex
   - getTapeDrive
   - getTransaction

<table>
<thead>
<tr>
<th>Method</th>
<th>Script Views</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDependency</td>
<td>View Z</td>
</tr>
<tr>
<td>getDiskDrive</td>
<td>View #</td>
</tr>
<tr>
<td>getMQDetail</td>
<td>Views 5UZ, 5VZ and 5NZ (requested as View 5NUVZ) — Requires MainView for WebSphere MQ product to be installed.</td>
</tr>
<tr>
<td>getSoftware</td>
<td>Views 0Z, 1Z, 2Z, 3Z, 4Z, 5Z, 6Z and 7Z (requested as View 01234567Z)</td>
</tr>
<tr>
<td>getStorageSubsystem</td>
<td>View -</td>
</tr>
<tr>
<td>getSysplex</td>
<td>View Z</td>
</tr>
<tr>
<td>getTapeDrive</td>
<td>View -@</td>
</tr>
<tr>
<td>getTransaction</td>
<td>View 0YZ — Requires MainView for CICS and MainView for IMS products to be installed.</td>
</tr>
</tbody>
</table>

*indicates methods that must succeed for a Mainframe to be created.

Though the DB2/IMS/MQ/CICS servers will be discovered, this detailed data will *not* be available in a z/OS Discovery Agent *only* install.

**Credentials**

To discover a mainframe computer you must have a z/OS user ID and password pair that is also defined in RACF, ACF2, Top Secret, or other external security managers. You configure the credential in the Management Systems tab in the Discovery Credentials page.

For further information on mainframe credentials, see Configuring mainframe credentials.

For information on configuring security for the BMC Discovery for z/OS Agent, see the BMC Discovery for z/OS Agent Installation Guide.

**Enabling additional mainframe methods**

You can view the methods for mainframe discovery by clicking Mainframe Discovery in the Discovery Platforms page. The following mainframe methods are disabled by default as they are not required for a basic discovery:

- getDatabaseDetail
- getDiskDrive
- getTapeDrive
- getTransactionProgram

**Enabling the remaining mainframe discovery methods**

To enable the remaining discovery methods:

1. From the Discovery section of the Administration tab, select Platforms.
2. Select Mainframe.
3. Select the Enable link in the following rows:
3. 
4. From the Discovery tab, select **Knowledge Management**.
5. Click **Create Pattern**.
6. From **Pattern Templates**:
   a. For the `getMQDetail` method, select the `template_mainframe_mq` template.
   b. For the `getDiskDrive` and `getTapeDrive` methods, select the `template_mainframe_storage` template.
   c. For the `getTransaction` method, select the `template_mainframe_transaction` template.
7. Edit the patterns according to the instructions contained in the comments. You need to add a single trigger for each pattern contained in the template. See **Pattern templates** for more information on using pattern templates.

⚠️ **getDiskDrive and getTapeDrive with CMDB Synchronization**

When enabling `getDiskDrive` and `getTapeDrive` in a system where CMDB synchronization is used, you should be aware that these methods cause a large number of CIs to be created.

ℹ️ You can view and download an example pattern module created from the templates [here](#).

**Mainframe discovery and CMDB synchronization**

For versions of BMC Atrium CMDB earlier than 7.6.03, you must apply the mainframe extension, `extension-703-800-Mainframe.zip` to use CMDB synchronization with mainframe discovery. For instructions about setting up CMDB synchronization and applying BMC Atrium CMDB extensions, see **Applying the ADDM Integration Extensions**.

**Mainframe template example**

ℹ️ **File Download**

The file can be downloaded from [here](#).

// This module contains example patterns based on the Mainframe pattern//

```
metadata  origin := "ONLINE_DOCS";end metadata;
from SupportingFiles.Mainframe.Support import MainframeModel 1.0;
```
pattern Mainframe_TapeDrive 1.0'''' This pattern creates a Storage node for TapeDrives. Overview tags IBM, TapeDrive; end overview;

triggers on dtd := DiscoveredTapeDrive; end triggers;
body MainframeModel.createTapeStorage(dtd); end body;
end pattern;

pattern Mainframe_DASD 1.0'''' This pattern creates a Storage node for DASDs beginning with AL. Overview tags IBM, DASD; end overview;

triggers on dasd := DiscoveredDiskDrive where volume_id matches regex '^AL';
body MainframeModel.createDiskStorage(dasd); end body;
end pattern;

pattern Mainframe_Transaction 1.0'''' This pattern creates a Detail node for Transactions beginning with CICSTRAN-C. Overview tags IBM, Transaction; end overview;

triggers on trn := DiscoveredTransaction where name matches regex '^CICSTRAN-C';
body MainframeModel.createTransaction(trn); end body;
end pattern;

pattern Mainframe_MQ 1.0'''' This pattern creates a Detail node for MQTransactions beginning with MQCHNL-. Overview tags IBM, MQ; end overview;

triggers on dmq := DiscoveredMQDetail where name matches regex '^MQCHNL-';
body MainframeModel.createMQDetail(dmq); end body;
end pattern;

Mainframe discovery methods

Some of the detail data that the z/OS Discovery Agent collects is dependent on additional MainView products being installed. The following table shows the MainView products used to obtain additional detail information. See the BMC Discovery for Z/OS documentation for further information.

<table>
<thead>
<tr>
<th>Method</th>
<th>Script Views</th>
</tr>
</thead>
<tbody>
<tr>
<td>getMainframeInfo</td>
<td>View Z</td>
</tr>
<tr>
<td>getMFPart</td>
<td>View Z</td>
</tr>
<tr>
<td>getApplication</td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>Script Views</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>getCouplingFacility</td>
<td>View Z</td>
</tr>
<tr>
<td>getDatabase</td>
<td>Views 1IZ and 1XZ (requested as View 1IXZ) — Requires MainView for DB2 and MainView for IMS products to be installed.</td>
</tr>
<tr>
<td>getDatabaseDetail</td>
<td>Views 1IJZ, 1ILZ and 1IKZ (requested as View 1IJKLZ) — Requires MainView for DB2 product to be installed.</td>
</tr>
<tr>
<td>getDependency</td>
<td>View Z</td>
</tr>
<tr>
<td>getDiskDrive</td>
<td>View #-</td>
</tr>
<tr>
<td>getMQDetail</td>
<td>Views 5UZ, 5VZ and 5NZ (requested as View 5NUVZ) — Requires MainView for WebSphere MQ product to be installed.</td>
</tr>
<tr>
<td>getSoftware</td>
<td>Views 0Z, 1Z, 2Z, 3Z, 4Z, 5Z, 6Z and 7Z (requested as View 01234567Z)</td>
</tr>
<tr>
<td>getStorageSubsystem</td>
<td>View -</td>
</tr>
<tr>
<td>getSysplex</td>
<td>View Z</td>
</tr>
<tr>
<td>getTapeDrive</td>
<td>View -@</td>
</tr>
<tr>
<td>getTransaction</td>
<td>View 0YZ — Requires MainView for CICS and MainView for IMS products to be installed.</td>
</tr>
</tbody>
</table>

*indicates methods that must succeed for a Mainframe to be created.

**Configuring mainframe credentials**

The **Mainframe** credentials tab is a holder for the mainframe credentials. This tab enables you to view, edit, add, and test mainframe credentials.

**Viewing mainframe credentials**

To view the mainframe credentials, perform the following:

1. From the Discovery home page, click **Credentials**.
2. Click **Management Systems**.
3. Click **Mainframe**.

The mainframe credentials page displays credentials checked in sequence, and the first matching entry is used. After a working mainframe credential is found, further credentials are not checked. To reorder mainframe credentials, drag the credential to the required position in the list.

The mainframe credentials are shown in color-coded boxes. The colors represent the level of login success achieved with that credential:

- Green: 100% success rate.
- Yellow: partial success.
- Blue: the credential has never been used.
- Red: 0% success rate.
The following information is shown for each credential:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential link</td>
<td>This is the first part of the heading link for the credential. The range of IP addresses on which this credential is intended to be used. A link is also provided showing the last successful use of the credential. This links to the Discovery Access for that use.</td>
</tr>
<tr>
<td>Description</td>
<td>A free text description of the credential supplied by the user who created the credential.</td>
</tr>
<tr>
<td>Usage</td>
<td>A summary of the success rate when the credential has been used, information about failures, and links to DiscoveryAccesses, credential lists and other useful diagnostic pages.</td>
</tr>
<tr>
<td>Actions</td>
<td>A list with the following options:</td>
</tr>
<tr>
<td></td>
<td>• Edit — To edit the credential, select Edit. See #To configure mainframe credentials below.</td>
</tr>
<tr>
<td></td>
<td>• Disable — To disable a credential, select Disable. The credential is a marked as disabled in the credential list. When a credential is disabled, this option is replaced with an Enable option. To enable the credential, click Enable.</td>
</tr>
<tr>
<td></td>
<td>• Delete — Delete the credential.</td>
</tr>
<tr>
<td></td>
<td>• Test — Select this to test the credential. See Testing mainframe credentials below for more information.</td>
</tr>
<tr>
<td></td>
<td>• Move to top — Moves the credential to the top of the list.</td>
</tr>
<tr>
<td></td>
<td>• Move to bottom — Moves the credential to the bottom of the list.</td>
</tr>
</tbody>
</table>

To configure mainframe credentials

To discover a mainframe computer you must have a z/OS user ID and password pair that is also defined in RACF, ACF2, Top Secret, or other external security managers. You configure the credential in the Management Systems tab in the Discovery Credentials page.

To add or edit a credential for a mainframe computer:

1. From the mainframe credentials page, perform one of the following actions:
   a. To add a new credential, click Add.
   b. To amend an existing credential, click Actions => Edit.
2. Enter the following information:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching Criteria</td>
<td>The IP address or addresses against which to use this mainframe credential. Select &quot;Match All&quot; to match all endpoints. Deselect it to enter IP address information which determines whether this credential is suitable for a particular endpoint. Enter IP address information in one of the following formats:</td>
</tr>
<tr>
<td></td>
<td>• IPv4 address (for example 192.168.1.100). Labelled v4.</td>
</tr>
<tr>
<td></td>
<td>• IPv4 range (for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*). Labelled v4.</td>
</tr>
</tbody>
</table>

⚠️ The following address type cannot be specified

• IPv4 multicast addresses (224.0.0.0 to 239.255.255.255)
As you enter text, the UI divides it into pills, discrete editable units, when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. Invalid pills are labeled with a question mark. You can also paste a list of IP addresses or ranges into this field. If any pills are invalid, a message stating the number of invalid pills is displayed above the range field. Clicking the link applies a filter which shows only invalid pills which you can then edit or delete. The filter can be removed by clicking clear in the Showing n of n label below the Range field. There is no paste option on the context sensitive (right click) menu.

WARNING: You cannot paste a comma-separated list of IP address information into the Range field in Firefox. This can crash the browser. You can use a space separated list without any problems.

- To edit a pill, click the pill body and edit the text.
- To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
- To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.

Pills are not currently supported in Opera.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>A check box enabling you to enable or disable the credential.</td>
</tr>
<tr>
<td>Username</td>
<td>Enter the z/OS username with which to log in to the mainframe computer.</td>
</tr>
<tr>
<td>Set password</td>
<td>When updating a credential, the password is shown as a series of asterisks in this field and it cannot be edited. To enter a new password, select the check box. The password entry field is cleared. Enter the password into the password entry field; the password text is not echoed to the screen.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a free text description of the credential.</td>
</tr>
<tr>
<td>Custom Mainframe Host Server Port</td>
<td>The port to use to connect to the mainframe. By default this is 3940. To use a different port, select the Enable custom mainview port? check box and choose a port number from the list. The list is populated with port numbers specified on the Discovery Configuration page.</td>
</tr>
</tbody>
</table>

3. To save the details, click **Apply**.

**Testing mainframe credentials**

**Recommendation**

You should confirm that the BMC Discovery for z/OS Agent is installed and functioning correctly with the credential testing tool before attempting discovery of the mainframe as the test includes a number of diagnostics. Problems with configuration are much simpler to resolve using this approach.

To test an existing mainframe credential:

1. From the mainframe credential list, select Actions \=> Test.
2. Enter the IP address for which you want to test credentials.
3. Click **Test**.

   When the test completes, the Management System Credentials page is refreshed with the result of either success or a failure. By clicking through the state link, you can view the results of the credential test in more detail.

   ![Image of the credential test result page]

   **This screen illustrates how to view more detailed results of the credential test.**

The credential test result page shows the discovery methods enabled in BMC Atrium Discovery, whether corresponding views are detected in the Discovery for z/OS agent, and the z/OS agent version.

For information on configuring security for the BMC Discovery for z/OS Agent, see the *BMC Discovery for z/OS Agent Installation Guide*.

### Discovering IBMi

BMC Atrium Discovery enables SNMP discovery of IBMi (AS/400) targets. IBMi discovery consists of a UDP request from an ephemeral port on the appliance to port 161 on the target. The IBMi system responds using the same ephemeral port that initiated the request. BMC Atrium Discovery listens for a response on only that port.

#### IBM i discovery methods

IBM i (AS/400) discovery is undertaken using SNMP. SNMP discovery is supported for ALL devices with an accessible SNMP agent. Discovery supports SNMP v1, v2c and v3. For some older platforms (for example Netware) the use of SNMP v1 might be required. This is defined on a per credential basis. Only read (GET, GETNEXT, GETBULK) access is required.

This page shows the method used, MIB values, and OIDs used.

<table>
<thead>
<tr>
<th>Method</th>
<th>MIB Values</th>
<th>OID</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDeviceInfo*</td>
<td>SNMPv2-MIB::sysDescr.0</td>
<td>1.3.6.1.2.1.1.1.0</td>
</tr>
<tr>
<td></td>
<td>SNMPv2-MIB::sysName.0</td>
<td>1.3.6.1.2.1.1.1.5.0</td>
</tr>
<tr>
<td>getHostInfo*</td>
<td>HOST-RESOURCES-MIB::hrSystemUptime.0</td>
<td>1.3.6.1.2.1.25.1.1.0</td>
</tr>
<tr>
<td></td>
<td>HOST-RESOURCES-MIB::hrMemorySize.0</td>
<td>1.3.6.1.2.1.25.2.2.0</td>
</tr>
<tr>
<td>getInterfaceList*</td>
<td>IP-MIB::ipAddrTable [ ipAdEntAddr, ipAdEntIfIndex, ipAdEntNetMask ]</td>
<td>1.3.6.1.2.1.4.20.1 [ .1, .2, 3 ]</td>
</tr>
<tr>
<td></td>
<td>IF-MIB::ifTable [ ifIndex, ifDescr, ifSpeed, ifPhysAddress ]</td>
<td>1.3.6.1.2.1.2.2.1 [ .1, .2, .5, .6 ]</td>
</tr>
<tr>
<td>getNetworkConnectionList</td>
<td>TCP-MIB::tcpConnTable [ tcpConnState, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, tcpConnRemPort ]</td>
<td>1.3.6.1.2.1.6.13.1 [ .1, .2, .3, .4, .5 ]</td>
</tr>
<tr>
<td></td>
<td>UDP-MIB::udpConnTable [ udpLocalAddress, udpLocalPort ]</td>
<td>1.3.6.1.2.1.7.5.1 [ .1, .2 ]</td>
</tr>
</tbody>
</table>
**Discovering WebSphere**

IBM WebSphere is an Application Server, otherwise known as a Web Container, developed by IBM.

WebSphere implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server, in addition it adds tools for configuration and management which can be configured through a web interface, and includes its own internal HTTP server.

**Discovering WebSphere**

BMC Atrium Discovery uses the IBM.WebSphereApplicationServer pattern to discover the WebSphere instance, identify the installation directory and determine the WebSphere version. The pattern extracts information from the WebSphere configuration files to create and populate a WebSphere SI and its attributes.

Extended WebSphere discovery is enabled by the activation of the WebSphere. ExtendedDiscovery.DiscoverWebSphere pattern. This is activated by default in a new installation of BMC Atrium Discovery, but in an upgraded appliance, it must be activated manually. The WebSphere.ExtendedDiscovery.DiscoverWebSphere pattern triggers on the creation or update of a WebSphere SI and performs a deep discovery of WebSphere instances via configuration files. It then parses the results to build a J2EE inferred model of its Applications and Resources.

Creation of a JDBC Resource Detail node triggers the CreateJDBCToDatabaseSI pattern which searches the BMC Atrium Discovery model for an SI representing that database. If the database SI is found the pattern creates relationships between the WebSphere Application Server SI and the nodes representing the database (see below). If the host that the database runs on has not been scanned, no further work is undertaken.

Creation of a J2EEApplication Component node does not trigger any further operations.

The pattern attempts to:

- Get the application names from serverindex.xml and create the application Software Component node.
- Get the JNDI names of all the resources used by each application from the `<appname.ear>/WEB-INF/web.xml` file
• Read the resources.xml files for the server, node and cell and get
database and mail resource info.
• Create the database and mail Detail nodes.
• Link each resource with the apps that use it.

The WebSphere.ExtendedDiscovery.DiscoverWebSphere pattern is fully described in
Configipedia.

Requirements for a full discovery
For a full discovery of WebSphere, the database or databases that it is using, and the relationships
between them, you need the following credentials:

• WebSphere host login credentials
• Database host login credentials
• Database credentials

Database nodes and relationships
The relationships created depend on the way that the database type is represented. (For example,
an Oracle database is represented by an SI whereas a MySQL database server is represented as
an SI and the individual databases by Detail nodes with contained by relationships to the database
server SI).

The use of detail nodes is not intended to be permanent.

WebSphere Discovery results
The following screen shows a successfully scanned IBM WebSphere instance.

The attributes section of the WebSphere SI contains sections for Components, and Details.

• Components are J2EE applications on WebSphere.
• Details are resources on WebSphere. These might be application specific or globally
  defined. They can be one of the following:
  • User Database Resource: the default WebSphere user management database.
  • Java Mail Resources
Discovering IBM Power Systems

IBM Power Systems are Power Architecture-based servers that run AIX or Linux operating systems. Power System discovery extends the discovery of the AIX LPARS and WPARS (kernel virtualization) and Linux LPARS to include dependencies between applications running on LPARs and physical networks and storage that they rely on.

In a Power System, the frame (modeled as a Host Container node) contains one or two Virtual I/O (VIO) LPARs and a number of host LPARs. The frame also contains a hypervisor that enables direct communication between LPARs. The VIO LPAR controls host LPARs access to the host container resources (disk, I/O) and acts as a network bridge between the internal VLAN (on which the VIO server is connected) and the external physical network.

Multiple frames can be managed by a Hardware Management Console (HMC) a separate host that is used to configure and manage logical partitions, move running LPARS within a frame, and workload sharing and balancing.

⚠️ In some early versions of HMC, hmcbash shell does not exist. HMC hosts which do not have hmcbash installed cannot correctly interpret the Discovery commands and are therefore not supported by BMC Atrium Discovery.

IBM Virtual I/O (VIO) Server discovery

The Virtual I/O (VIO) Server is implemented as a management software layer that runs on top of an AIX 5.3 OS instance. The VIO management layer is not used for discovery; rather, the underlying AIX instance is used. The VIO management layer provides command line access using the ioscli command, which is a limited and restricted shell environment. The restrictions on users of this shell make it unsuitable for discovery by BMC Atrium Discovery (for example, key commands are unavailable and shell I/O redirection is not permitted).

In order to discover a VIO server you must add a normal UNIX user to the underlying AIX 5.3 instance, which requires root access. To obtain root access, you must either log into the AIX 5.3 instance directly as the root user (if such access has been enabled, it is not enabled by default) or log in to the VIO restricted shell and "escape" to the "OEM environment" using the following command:

```
$ oem_setup_env
```

After you have root access, you can add the discovery user using the useradd command as you would for any other AIX host:
Before you can use this account you need to configure a password.

```
# passwd discovery
```

You are prompted twice to enter a new password.

After the user is configured and a corresponding credential added to BMC Atrium Discovery, VIO Servers are discovered using the standard AIX discovery scripts. A discovered VIO server is shown in the following screen.

⚠️ It is also possible to avoid adding the local UNIX user account by escaping to the OEM environment in the standard AIX discovery script. However, this would give BMC Atrium Discovery unrestricted root access to the OS. This would also affect normal AIX hosts (where the `oem_setup_env` command does not exist). However, in this case you might only receive a runtime error message that you can ignore. It is not possible to detect the presence of `oem_setup_env` on the VIO server due to the shell restrictions.

**Hardware Management Console (HMC) discovery**

The Hardware Management Console consists of a dedicated host with application software that runs on a Linux OS instance. The HMC layer provides command line access, but only in a limited and restricted shell environment. Unlike the VIO Server, this environment is sufficient for discovery by BMC Atrium Discovery with the following caveats:

- No process information is returned. There is no HMC command to report processes running on the HMC host. Because the HMC is a dedicated host running nothing but the HMC application, this should not be an issue.
- No port to process information. There is no HMC command to report this information. The `lsof` command would be used on other platforms.
- No HBA information is available. There are no HMC commands to report HBA information. The HMC hardware does not use an HBA.

The user account used to discover HMC requires the `HMCVIEWER` role.

The HMC management software SI contains the details on the hosts, host containers, and Power Systems that the HMC is managing. The Power Systems are represented as Detail nodes, as illustrated in the following screen.

**Example visualization**

The following screen shows a visualization of an HMC management software SI with links to the Host Containers (frames) managed by the HMC. Contained within each of the frames is a single VIO server and a number of LPARS. To view this visualization, select Management from the Visualizations menu on the HMC management software SI.

**This diagram illustrates a HMC management software SI visualization.**
Discovering Cisco Nexus switch IP addresses

To get information about all the IP addresses in each VRF IP table, you must configure VRF contexts for each VRF. To do so, use the following configuration command in the Cisco Nexus switch console:

```
snmp-server context "name of the context" vrf "name of the associated vrf"
```

For example, if the VRF name is “RED”, to create its associated context, the command will be:

```
snmp-server context REDCONTEXT vrf RED
```

For more information about Cisco Nexus switches, visit Cisco documentation and browse Products > Switches > Data Center Switches. From the list, select the required Cisco Nexus switch series and the corresponding switch. If you want to know about the network devices that can be discovered by BMC Atrium Discovery, see Network devices.

Improving discovery

BMC Atrium Discovery provides as much information as it can about your environment. To maximize this ability, there are features in the product to help you improve the level and quality of access to the environment.

- **Monitoring credential usage**: Discovery relies on access credentials. The credentials management tools provide an overview of the success rate of the credentials to assist you in monitoring the roll out and currency of the credentials in use. This can monitor Login Credentials, Windows proxies and SNMP Credentials.
- **DiscoveryAccess page**: Discovery records a lot of useful metadata information about each access to the environment. This page describes some of the information available and how to use the results to troubleshoot access issues.
- **Discovery Conditions**: There are many scenarios where the system can detect that data could be improved. The Discovery Condition tools enable you to see advice on what actions you can take, and which hosts are impacted by these actions.

Monitoring credential usage

On the Credentials pages, credentials are shown in color coded boxes. The colors represent:

- **Green** — 100% success rate
- **Yellow** — partial success
- **Blue** — the credential has never been used
1. **Red** — 0% success rate

Where a credential has been used successfully, a link showing the number of successful uses is displayed. If there is a single use then it links to the DiscoveryAccess page for that credential use. If there are multiple uses, it links to a list of DiscoveryAccess pages. Where a credential has failed periodically but has been used successfully, a link is provided to the last successful Discovery Access, and links are also provided to a list of Session Results for failure cases.

Where a credential has never been used successfully there is a link to the SessionResult page for that attempted access.

---

### Credential Windows proxy note

The Credential Windows proxy uses credentials defined on the Login Credentials page, so the usage statistics display in the Login Credentials page, rather than the Windows proxy Management page.

---

### DiscoveryAccess page

A DiscoveryAccess is a single access to a discovery endpoint. When an endpoint is scanned, a Discovery Access node is created which records information on the interaction that BMC Atrium Discovery has with that endpoint. When BMC Atrium Discovery is unable to access a host, the DiscoveryAccess is a good starting point for troubleshooting. If a DiscoveryAccess is about to be deleted as part of DDD aging, a red banner stating “This node will be removed shortly as part of DDD aging”.

**To view a DiscoveryAccess page**

You can view DiscoveryAccesses from a number of places in the user interface.

**From a host node**

From the view page for a host node:

1. Scroll down to the Inference section.
2. Click the main link in that section (for example: **172.17.3.116 SUCCESS 17/102014 10:48 - 17/10/2014 10:55**)
   
   The Example DiscoveryAccess is shown.

**From the Discovery Recent Runs page**

From the Recent Runs tab of the Discovery Status page:

1. Click a discovery run.
2. Scroll down to the Endpoint field.
3. Click the **DiscoveryAccess** link.
   If it is a link to a single DiscoveryAccess, then that **Example DiscoveryAccess** is shown. If there are multiple DiscoveryAccesses, then a list page is displayed.
4. Click a DiscoveryAccess line to view the **Example DiscoveryAccess**.

**Example DiscoveryAccess**

The following screens show DiscoveryAccess pages for a UNIX host, a Windows host, a network device, and a mainframe computer. The Windows example shows multiple credentials. One is the host login credential and the other is the WBEM credential used to access (in this case) some storage management software running on the host.

This screen illustrates a DiscoveryAccess page for a UNIX host.

This screen illustrates a DiscoveryAccess page for a Windows host.
This screen illustrates a DiscoveryAccess page for a network device.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Section</td>
<td>The endpoint (IP address) scanned in this discovery access.</td>
</tr>
<tr>
<td>Start Time</td>
<td>The time at which the scan started.</td>
</tr>
<tr>
<td>End Time</td>
<td>The time at which the scan finished.</td>
</tr>
<tr>
<td>Total Duration</td>
<td>The time it took to discover and process the data (Start Time to End Time).</td>
</tr>
<tr>
<td>Discovery Run</td>
<td>A link to the Discovery Run that this DiscoveryAccess is part of.</td>
</tr>
<tr>
<td>Previous Discovery Access</td>
<td>A link to the previous DiscoveryAccess with the same endpoint. It is not displayed if it is the first in a list.</td>
</tr>
<tr>
<td>Next Discovery Access</td>
<td>A link to the next DiscoveryAccess with the same endpoint. It is not displayed if it is the last in a list.</td>
</tr>
</tbody>
</table>

This screen illustrates a DiscoveryAccess page for a mainframe computer.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Discovery Access</td>
<td>A read-only summary showing the node kind, OS type, and version.</td>
</tr>
<tr>
<td>Device Summary</td>
<td>A link to the inferred entity that was created or updated as part of the scan. It is not displayed if nothing was created or updated.</td>
</tr>
<tr>
<td>Inferred Entity</td>
<td>A link to the inferred entity that was created or updated as part of the scan. It is not displayed if nothing was created or updated.</td>
</tr>
<tr>
<td>Status Section</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>The current state of the DiscoveryAccess. This can be Started or Finished.</td>
</tr>
<tr>
<td>End State</td>
<td>The end state of the discovery run. This can be one of the following:</td>
</tr>
<tr>
<td></td>
<td>Successful discovery results in GoodAccess; during processing you might see first DeviceIdentified and later HostInferred.</td>
</tr>
<tr>
<td></td>
<td>For optimised discovery, that is, started but stopped for a reason, you might see Opt1stScan, OptNotBestIP, OptRemote, OptAlreadyProcessing.</td>
</tr>
<tr>
<td></td>
<td>For unsuccessful discovery you might see NoResponse, UnsupportedDevice, NoAccess, Excluded, Error.</td>
</tr>
<tr>
<td></td>
<td>See the table below for information on how the end state and result relate to the discovery scenario.</td>
</tr>
<tr>
<td>Result</td>
<td>The result of the DiscoveryAccess. This might be Skipped, NoResponse, Success, NoAccess, or Error.</td>
</tr>
<tr>
<td></td>
<td>See the table below for information on how the end state and result relate to the discovery scenario.</td>
</tr>
<tr>
<td>Errors</td>
<td>If there were any errors detected by the ECA engine during discovery this will link to those errors. Examples are:</td>
</tr>
<tr>
<td></td>
<td>• Error detected by the ECA engine: This is typically an internal rule error or pattern error. Usually triggered when the data returned does not match what is expected.</td>
</tr>
<tr>
<td></td>
<td>• Unable to get the deviceInfo: ExecutionFailure: Discovery has attempted to run a command but a failure has been reported.</td>
</tr>
<tr>
<td></td>
<td>• Unable to get the deviceInfo: NoAccessMethod: No access method, this is frequently because it is not meaningful, such as getting patches on Linux.</td>
</tr>
<tr>
<td></td>
<td>• Unable to get the deviceInfo: NoSuchDevice: There is a NoResponse endstate. That is, nothing detected on the IP address.</td>
</tr>
<tr>
<td></td>
<td>• Unable to get the deviceInfo: TRANSIENT Unable to get the deviceInfo: TRANSIENT_CallTimedout: Probably caused by the reasoning timeout; the call to discovery is taking too long to complete.</td>
</tr>
<tr>
<td></td>
<td>• Unable to get the deviceInfo: UNKNOWN: Some other CORBA error. Contact Customer Support.</td>
</tr>
<tr>
<td>Session Results</td>
<td>If there were any failures attempting to get a session on the endpoint this will link to a list of failures and successes. See below for details.</td>
</tr>
<tr>
<td>Discovery Details Section</td>
<td></td>
</tr>
<tr>
<td>Data Source</td>
<td>Does this Discovery Access originate from this appliance, come from a scanning file, or was it consolidated from a scanning appliance.</td>
</tr>
<tr>
<td>Credentials Used</td>
<td>A link to the Windows proxy, credential or credentials used in this Discovery Access. The link name is a hash of details of the credential; it does not provide the credential itself. You are not shown the Credential pages if you do not have permissions to view them. This field is not displayed as a link on the consolidation appliance for scans which have been consolidated from a scanning appliance.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Discovery Start Time</td>
<td>The time at which discovery started on the scanning appliance. This field is only displayed on the consolidation appliance for scans which have been consolidated from a scanning appliance.</td>
</tr>
<tr>
<td>Discovery End Time</td>
<td>The time at which discovery completed on the scanning appliance. This field is only displayed on the consolidation appliance for scans which have been consolidated from a scanning appliance.</td>
</tr>
<tr>
<td>Session Establishment Duration</td>
<td>The time it took to establish the session, that is, to log onto the host.</td>
</tr>
<tr>
<td>Total Discovery Duration</td>
<td>The time taken to establish a session and run commands.</td>
</tr>
<tr>
<td>On Hold Since</td>
<td>If the discovery is currently paused, the time at which it was paused.</td>
</tr>
<tr>
<td>On Hold Duration</td>
<td>If the discovery has been paused, the elapsed time since it was paused.</td>
</tr>
<tr>
<td>Skipped Entity</td>
<td>A summary of the entity which was skipped.</td>
</tr>
<tr>
<td><strong>Standard Discovery</strong></td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>The discovery method used. The methods available on each platform are shown on the following pages:</td>
</tr>
<tr>
<td></td>
<td>- UNIX and related operating systems</td>
</tr>
<tr>
<td></td>
<td>- Windows operating systems</td>
</tr>
<tr>
<td></td>
<td>- Mainframe</td>
</tr>
<tr>
<td></td>
<td>- IBM i</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the discovery access for the method. This is OK or the failure reason.</td>
</tr>
<tr>
<td>Script</td>
<td>The name of the script used, if any.</td>
</tr>
<tr>
<td>Access</td>
<td>The access method used to connect to the endpoint (for example, ssh, telnet, rlogin, and so on).</td>
</tr>
<tr>
<td>Result</td>
<td>A link to the node or nodes created by this discovery method.</td>
</tr>
<tr>
<td><strong>Additional Discovery</strong></td>
<td></td>
</tr>
<tr>
<td>Method</td>
<td>The additional discovery method used. These are discovery methods called by patterns, for example:</td>
</tr>
<tr>
<td></td>
<td>- getFileInfo</td>
</tr>
<tr>
<td></td>
<td>- getFileMetadata</td>
</tr>
<tr>
<td></td>
<td>- runCommand</td>
</tr>
<tr>
<td></td>
<td>- getDirectoryListing</td>
</tr>
<tr>
<td></td>
<td>- wbemEnumInstances</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the discovery access for the method. This is OK or the failure reason summarized into links.</td>
</tr>
<tr>
<td>Script</td>
<td>The name of the script used.</td>
</tr>
<tr>
<td>Access</td>
<td>The access method used to connect to the endpoint. For example: ssh, telnet, rlogin, and so forth.</td>
</tr>
<tr>
<td>Result</td>
<td>A link to the node or nodes created by this discovery method.</td>
</tr>
</tbody>
</table>
**DiscoveryAccess state**

The following table shows the possible discovery scenarios and the resulting `end_state` and `result` attributes of the Discovery Access node:

In addition to `end_state` and `result` attributes on the DiscoveryAccess there is also a `reason` attribute that contains further details. There is no fixed set of values for `reason`.

The following state diagram and table might be of use understanding the results of an attempted access.

---

**Discovery Scenario** | **Resulting State of DiscoveryAccess**
--- | ---
IP Injected In Exclude list  
*Example:* The user has requested a scan of an IP that is in an exclude range.  | `end_state=Excluded`  
`result=Skipped`

IP Injected IP Response Desktop host  
*Example:* The user has requested a scan of an IP that is a desktop host when scanning of desktop hosts has been disabled. The reason attribute is populated with Desktop host discovery has been disabled.  | `end_state=Excluded`  
`result=Skipped`

IP Injected Already Processing this IP  
*Example:* The user has requested a scan of an IP that is currently being processed as part of an earlier scan.  | `end_state=OptAlreadyProcessing`  
`result=Skipped`

IP Injected Second Scan Optimization (Best IP)  
*Example:* The user has requested a scan of an IP that is on a host that has already been scanned using another IP address that is considered to be the best IP to use to scan that host.  | `end_state=OptNotBestIP`  
`result=Skipped`

IP Injected No IP Response  
*Example:* No response was received from the scanned IP address. The IP address is considered dark space, and no DiscoveryAccess node is created.  | `end_state=UnsupportedDevice`

IP Injected IP Response  
*Example:* After a sweep scan of an IP address, Discovery has received a response. No identification has taken place other than that there is a device of some description that has responded to the sweep scan. If the device is subsequently not recognized then the `end_state` is set to `UnsupportedDevice` as below.  | `end_state=DeviceIdentified`  
`result=Success`

IP Injected IP Response Device Type not supported  
UnsupportedDevice is the default `end_state` for any device that BMC Atrium Discovery can detect, but do not have any knowledge of in Reasoning to build an inferred node. We might still be able to identify what the device is and some key properties, or we might know nothing more than the fact it returned a ping.  | `end_state=UnsupportedDevice`  
`result=Skipped`

This screen illustrates the result of an attempted access of a Discovery Access node.
Troubleshooting using session results

There are three scenarios that are highlighted using Session Results:

- Initial scan and successful connection: A number of unsuccessful connection attempts followed by a successful connection shows BMC Atrium Discovery selecting credentials.
- Failure to connect: BMC Atrium Discovery was unable to connect. Typical problems could be failed credentials or poor network connectivity.
- Failure at start of scan before successful connection: This can occur when a credential fails (expiry) but a later credential is successful.

Checking credentials after a failure

1. From the Discovery Access, click the session results link.
2. Click the Connection timed out rows to display the SessionResult page for that specific login attempt.
   This page shows information on the attempted login, including a credential link and a link to the DiscoveryAccess. The credential link is a hash of the credential name which links to that credential in the Login Credentials page only if you have sufficient permissions to view the credentials.
3. Click the test button.
   If the credential tests successfully, it is likely that a transient network problem caused the connection time-out. Alternatively, you can navigate back to the Discovery Access page and rescan from the Discovery Actions menu.

<table>
<thead>
<tr>
<th>Discovery Scenario</th>
<th>Resulting State of DiscoveryAccess</th>
</tr>
</thead>
<tbody>
<tr>
<td>to identify the device further. This might mean that the OS is unrecognized, garbage has been returned or there is a shell error. This can be caused by a firewall. For more information, see the IP Fingerprinting information in the Base Device Discovery section.</td>
<td>end_state=NoAccess result=NoAccess</td>
</tr>
<tr>
<td>IP Injected IP Response No HostInfo recovered</td>
<td>end_state=NoAccess result=NoAccess</td>
</tr>
<tr>
<td>IP Injected IP Response No MACAddresses recovered</td>
<td>end_state=HostInferred result=Success</td>
</tr>
<tr>
<td>IP Injected IP Response HostInfo and MACAddresses recovered First Scan Optimization</td>
<td>end_state=Opt1stScan result=Skipped</td>
</tr>
<tr>
<td>IP Injected IP Response HostInfo and MACAddresses recovered First Scan Optimization not needed</td>
<td>end_state=GoodAccess result=Success</td>
</tr>
<tr>
<td>IP Injected Traceback captured</td>
<td>end_state=Error</td>
</tr>
<tr>
<td>IP Consolidated when originally optimized on Discovery Appliance</td>
<td>end_state=OptRemote result=Skipped</td>
</tr>
</tbody>
</table>
Discovery Conditions

Discovery Conditions are hints to administrators. They describe configuration issues in an estate that, if resolved, can help BMC Atrium Discovery retrieve more detailed or complete data. Doing this improves the view of the estate, so that users can act on the knowledge it gathers with greater confidence.

The following Discovery Conditions are described in Configipedia:

- **Discovery Condition AIX Netstat truncates IPv6 Addresses** (AIX host with out of date bos.net.tcp.client package)
- **Discovery Conditions Windows NIC** (One or more of speed, raw_speed, duplex or negotiation attributes missing from network interface card)
- **Discovery Condition Red Hat Enterprise Linux RHBA_2012_0188** (Red Hat host with out of date net-tools)
- **Discovery Condition Solaris 10 Command Line Truncation**
- **Discovery Condition Solaris 11 Command Line Truncation**
- **Discovery Condition Solaris 9 Command Line Truncation**
- **Discovery Condition Solaris 8 Command Line Truncation**
- **Discovery Condition vSphere Socket Exhaustion** (VMware ESX/ESXi Host with CVE-2011-1785 Vulnerability)
- **Discovery Condition Windows 2003 KB932370** (Windows 2003 without KB932370)
- **Discovery Condition Windows XP KB936235** (Windows XP without SP3 or KB936235)

Viewing all current Discovery Conditions

1. From the Discovery tab, select Discovery Reports.
2. In the Directly Discovered Data Index section, select Discovery Conditions.

**Discovery Conditions are not logged until a Host with the issue is scanned.**

Viewing all current Discovery Conditions from a host view

1. Scroll towards the bottom of the host view to find the Inference section.
2. If Discovery Conditions have been logged, they are summarized under the Discovery Conditions section, following Discovery Access section.

*This screen illustrates a summary displayed of logged Discovery Conditions.*
Viewing hosts with Discovery Conditions logged

The following section describes various methods of seeing which hosts have discovery conditions logged against them.

Discovery Condition Detailed List report

This report shows a list of hosts with logged Discovery Conditions along with the time the condition was first logged sorted so the most recent are at the top. Clicking through will show the Host page. This report is ideal for exporting to CSV to make a task list for resolving the issues.

1. From the Discovery tab, select Discovery Reports.
2. In the Discovery Dashboard Reports section, select Discovery Condition Detailed List.

This report is also available from the Discovery Dashboard.

Hosts impacted by Discovery Conditions report

This report shows a detailed list of hosts that have Discovery Conditions logged. This report is ideal for filtering the selection of Hosts on the detailed criteria to prioritize.

1. From the Discovery tab, select Discovery Reports.
2. In the Discovery Dashboard Reports section, select Hosts impacted By Discovery Condition.

This report is also available from the Discovery Dashboard.

Understanding Discovery Conditions

Discovery Conditions have the following fields:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The overview of the condition.</td>
</tr>
<tr>
<td>Impact</td>
<td>The impact of the condition.</td>
</tr>
<tr>
<td>BMC Link</td>
<td>Link to our online detailed description of the condition and advice on how best to resolve it.</td>
</tr>
<tr>
<td>Description</td>
<td>A short description of the condition</td>
</tr>
<tr>
<td>Resolution</td>
<td>A short summary of the resolution steps.</td>
</tr>
<tr>
<td>Vendor Links</td>
<td>Links to vendor descriptions and fixes if relevant.</td>
</tr>
<tr>
<td>Applies to</td>
<td>Link to all the Host impacted by this condition.</td>
</tr>
</tbody>
</table>

The following screen shows an example view of a Discovery Condition:
This screen illustrates an example view of a Discovery Condition.

**Monitoring appliance performance**

The appliance generates statistics on aspects of its operation and reports them in the appliance performance pages. Appliance performance tracks performance statistics for engines, patterns, and hardware.

- Appliance status
- Engine performance
- Pattern performance
- Hardware statistics
- Datastore cache performance
- DDD removal statistics

**Appliance status**

A dialog is displayed when you click the Appliance Status icon in the dynamic toolbox. See [Dynamic Toolbox](#). The appliance status page is displayed when you click the link in the dialog. The link description can have one of the following descriptions:

- **No Problems Detected** — The status is green. No problems have been detected.
- **Status Information Available** — The status is green, but at least one potential problem has been detected that has an information-level message.
- **Minor Problems Detected** — At least one minor problem has been detected with the appliance.
- **Major Problems Detected** — At least one major problem has been detected with the appliance.
- **Critical Problems Detected** — At least one critical problem has been detected with the appliance.

You can also access this page by clicking the **Administration** tab from the primary navigation bar and clicking the Baseline Status icon in the Appliance section.

One or more of the following actions can be configured to occur if the appliance status is reported as critical, major, or minor.

- Send Email
- Restrict Network Access
- Stop Discovery
The configuration checks that are performed for each item in the Appliance Status Page are described in Baseline Configuration.

The following screen illustrates the Appliance Baseline page and the various status states for each listed appliance.

Where an item has a report or configuration page (such as Appliance Specification or Appliance Configuration Files Tripwire) a hyperlink to that page or report is provided. Where no such page or report is available (such as Appliance Firewall), there is no hyperlink.

The following buttons are provided at the bottom of the page:

- **Check Baseline Now** — Performs an immediate check of the appliance against the baseline.
- **Update All Baselines** — Updates all checks in the baseline status to their current values.
- **Configure Actions ...** — Opens the Appliance Actions page. From here you can configure the actions you want to take in the event of baseline failures. See Configuring Actions.
- **Configure Options** — Opens the Appliance Baseline Options page. From here you can configure automatic email messaging on the event of a baseline failure, and individual network services to maintain if network access is restricted. See Configuring Options.
- **Cancel** — Closes the Appliance Baseline page.

**Engine performance**

In BMC Atrium Discovery each ECA engine provides information about how much work it is doing. This information is available when Reasoning is running at debug level as it logs information about the events passing through the system and rules that are being executed, however, this log is difficult for many people to understand. On the Appliance Performance page the work that ECA engine is doing is presented graphically.

To view the appliance performance page

1. Click the Administration tab from the primary navigation bar.
2. Click the Performance icon in the Appliance section.
   The Appliance Performance page displays the Engines tab.

The Engine Statistics chart shows the event queue size and number of events processed for the last 24 hours. The left hand margin represents midnight, the right hand margin represents 23.59 hours. You can use the drop-down selector to view the engine statistics chart for any of the last ten days.
The absolute values are useful for historical comparison, or comparison with other instances of BMC Atrium Discovery. They might be useful as supporting information when diagnosing problems.

**Pattern performance**

The Pattern performance page displays timing information about TPL pattern performance. TPL provides considerable power which can be used in discovering an environment. Unfortunately some patterns are inherently inefficient or perform poorly in specific customer environments.

Determining the performance of individual patterns is extremely difficult, involving running Reasoning at debug level and then piecing together all the rules run for an individual pattern. Each pattern has performance information gathered for it by Reasoning and this is presented graphically on the Appliance Performance page. This makes it easier for customers and BMC Customer Support to determine which patterns are performing badly.

- To access patterns
- To use the Pattern Performance page
- What to do if there is a problem

**To access patterns**

1. Click the **Administration** tab from the primary navigation bar.
2. Click the **Performance** icon in the Appliance section.
   - The Appliance Performance page displays the **Engines** tab.
3. Click the **Patterns** tab.

You can use the drop-down selector to view the pattern performance statistics chart for any of the last ten days. If dates are not available on the selector, no log has been created for those days.

Invocation and timing information (in seconds) is displayed for each pattern and is described in the following table.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern Name</td>
<td>The name of the pattern.</td>
</tr>
<tr>
<td>Invocations</td>
<td>The number of times that the pattern has been invoked in the reporting period.</td>
</tr>
<tr>
<td>Column name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Average Execution Time</td>
<td>The average execution time. This time excludes time spent waiting for discovery commands to complete. It is the sum of the time spent running searches in the data store, updating the model, updating inference relationships and any other work the pattern did including the execution of the pattern code itself.</td>
</tr>
<tr>
<td>Max Execution Time</td>
<td>The maximum amount of time taken to execute the pattern as defined above.</td>
</tr>
<tr>
<td>Min Execution Time</td>
<td>The minimum amount of time taken to execute the pattern as defined above.</td>
</tr>
<tr>
<td>Average Discovery Time</td>
<td>The average time this pattern has spent running discovery commands.</td>
</tr>
<tr>
<td>Average Search Time</td>
<td>The average time this pattern has spent performing searches on the data store.</td>
</tr>
<tr>
<td>Average Modeling Time</td>
<td>The average time this pattern has spent creating, updating or deleting nodes in the model.</td>
</tr>
<tr>
<td>Average Inferencing Time</td>
<td>The average time this pattern has spent creating, updating or deleting the relationships between DDD and inferred nodes.</td>
</tr>
</tbody>
</table>

To use the Pattern Performance page

You would typically view this page if you are concerned that the performance of BMC Atrium Discovery is being impacted by patterns or you are testing a new pattern that you have written. The key points to look for are spikes in the following:

- **Average Discovery Time** — A spike in the average discovery time might indicate credential or network problems.
- **Average Search Time** — Where this measure appears significantly larger than all others in the column it shows that this pattern is using the search service more than is usual. Depending on the pattern, and the job it is trying to do, this might indicate a problem. Where this is a problem, check the search statement to see whether it can be made more efficient.
- **Average Modeling Time** — Creating an excessive number of nodes in the model can increase the modeling time. Ensure that your pattern only creates the nodes that you actually need, for example do not needlessly delete and recreate nodes.
- **Average Inferencing Time** — Inferencing time is affected to a lesser extent by patterns than the search and modeling times. An increase here might indicate that the pattern creates an excessive number of nodes.

What to do if there is a problem

If the pattern page highlights one or more of the problems described previously, the course of action to take depends whether it is a TKU pattern or a pattern that you have developed in-house.
TKU patterns

TKUs are shipped each month. If a TKU pattern is causing a performance spike, then you should first check to see that you are using the most up to date TKU. If you are then you should contact Customer Support to report the issue who will advise on the best course of action.

In-house developed patterns

If you are using a pattern that you have developed, then you should consider the following common errors:

- Poor trigger — A trigger that is too wide and results in many invocations of a pattern which end up writing no data. This might be highlighted by a large mismatch in the number of invocations against the number of SIs managed by the pattern. You can find out how many SIs are being managed by a pattern in the Maintained Software Instances of the pattern’s page. As a rule of thumb it is two orders of magnitude more efficient to stop in the trigger rather than in the pattern body.
- Delay in stopping — Patterns are triggered and run and continue to process data until they complete or reach a condition which stops the pattern. Where a condition could cause the pattern to stop early, it should be placed as early as possible in the pattern to avoid wasted processing.
- Pattern doing too much work — Individual patterns should ideally perform one task, for example, creating a software instance. Examples of patterns doing too much work are:
  - Creating SIs and a BAI in a single pattern. Rather, you should break the pattern down so each pattern creates one type of node which triggers the next.
  - Loops containing discovery calls. You should try to make these calls once and then access the stored results as necessary from the loops.
  - Creating an SI with multiple detail nodes. Try to simplify or reduce the amount of detail stored.

Hardware statistics

This topic provides information and instructions for viewing hardware statistics:

- To access the the hardware statistics
- Daily SAR statistics
- Hourly SAR statistics
- Hourly memory usage statistics
- Hourly average traffic statistics
- Daily disk usage

To access the the hardware statistics

1. Click the Administration tab from the primary navigation bar.
2. Click the Performance icon in the Appliance section.
   The Appliance Performance page displays the Engines tab.
3. Click the **Hardware** tab.

An XML link is provided with each chart. Click this link to download the chart as XML.

**Daily SAR statistics**

The Daily SAR Statistics chart shows the CPU usage and disk IO wait time as percentages for the last 30 days. The left hand margin represents the start time until 30 days have elapsed; subsequently it represents 30 days in the past. The right hand margin represents the most recent data.

![Daily SAR Statistics chart](image)

**Hourly SAR statistics**

The Hourly SAR Statistics chart shows the CPU usage and disk IO wait time as percentages for the last 24 hours. The left hand margin represents midnight, the right hand margin represents 23.59 hours.

![Hourly SAR Statistics chart](image)

**Hourly memory usage statistics**

The Hourly Memory Usage Statistics chart shows the memory usage as a percentage for the last 24 hours. The left hand margin represents midnight, the right hand margin represents 23.59 hours.

![Hourly Memory Usage Statistics chart](image)
Hourly average traffic statistics

The Hourly Average Traffic Statistics chart shows the transmitted and received network traffic for each network interface for the last 24 hours. The left hand margin represents midnight, the right hand margin represents 23.59 hours.

Daily disk usage

The Daily Disk Usage chart shows the disk usage for each physical device and each logical RAID device, as a percentage of used disk capacity. The left-hand margin represents the start time until 30 days have elapsed; subsequently it represents 30 days in the past. The right-hand margin represents the most recent data.

This screen illustrates the Daily Disk Usage chart.

Datastore cache performance

The datastore makes use of an in-memory cache which is used for all read and write operations. The datastore stores data on disk in pages, and any system or user request to the datastore can require data from many pages. Any page not currently in the in-memory cache must be retrieved from disk. Every datastore request can result in a number of cache page hits or misses. In addition, if the data in a page has been modified since being read into to the cache, it is termed *dirty* and must be written to the disk before the page is removed from the cache. If a page is unchanged since being loaded into the cache, it is termed *clean* and can be unloaded without writing to disk.

If the datastore is spending a lot of time swapping pages in and out of the cache, it can cause performance problems.

- Datastore cache performance factors
Examples
To view the datastore cache performance page

Datastore cache performance factors
The Datastore cache performance page enables you to determine whether the cache is performing effectively. It shows the following data at ten minute intervals:

- Cache hits
- Cache misses
- Clean pages evicted
- Dirty pages evicted

Cache hits and cache misses are the most important indicators of cache performance.

Expected datastore performance depends a great deal on the environment being discovered, the performance of the appliance hardware and disk subsystem, and on how heavily the system is used for data browsing and reporting. The following figures are representative of a "typical" installation in a "typical" environment while actively scanning or consolidating.

- A datastore cache that is performing well can be expected to have cache hits in the order of tens of millions and misses in the order of tens of thousands or fewer per ten minutes.
- A datastore cache that is experiencing in the order of hundreds of thousands of misses per ten minutes might have adverse effects on the system as a whole.

In previous versions of BMC Atrium Discovery changing datastore cache settings required manual configuration, but since version 10 it is managed automatically.

Examples

This graph shows three discovery runs in which a datastore cache is performing well with cache hits between 75 and 95 million per ten minutes and cache misses below 45 thousand per ten minutes.
This graph shows a discovery run from midnight to 1am in which the datastore cache is experiencing just under 50 million cache hits per ten minutes, but with 375 thousand cache misses per ten minutes. The rate of cache hits is typical, though the rate at which cache misses are occurring is high and might be causing adverse affects to the system as a whole.

To view the datastore cache performance page

1. Click the **Administration** tab from the primary navigation bar.
2. Click the **Performance** icon in the Appliance section. The Appliance Performance page displays the **Engines** tab.
3. Click the **Datastore** tab.

**DDD removal statistics**

- Using the DDD removal statistics page
- Viewing DDD removal statistics

**Using the DDD removal statistics page**

The Directly Discovered Data (DDD) removal statistics page shows the total number of DiscoveryAccesses in the datastore and those eligible for removal. If DiscoveryAccess removal is keeping up with DiscoveryAccess creation, then the number of eligible Discovery Access nodes is zero, or near zero, and the continual DDD aging scheme is sufficient. See example 1 in the diagram below.

With the continual DDD aging scheme in operation, if the trend of eligible DiscoveryAccesses is rising over a two week period, you might have contention between the DDD creation and removal processes. See example 1 in the diagram below. In this scenario you might find that using DDD
removal blackout windows to schedule removal between the gaps in your discovery schedule might give you better overall system performance. Once you have set up DDD removal blackout windows you should continue to monitor the DDD removal statistics page to ensure that removal is keeping up with creation. See diagrams 3 and 4 in the diagram below.

1. DDD removal commences and a steady state achieved

2. DDD removal not keeping up

3. Blackout windows too long — DDD removal not permitted to keep up

4. Blackout windows providing sufficient time for DDD removal to keep up.

**Viewing DDD removal statistics**

1. Click the Administration tab from the primary navigation bar.
2. Click the Performance icon in the Appliance section.
   - The Appliance Performance page is displayed showing the Engines tab.
3. Click the DDD Removal tab.

The DDD removal statistics page shows the total number of DiscoveryAccesses in the datastore and those eligible for removal for the last 28 days.
This screen illustrates DDD Removal Statistic on the Appliance Performance page.

# Knowledge management

From BMC Atrium Discovery version 10, pattern and knowledge management have been improved and a new Knowledge management page is added, replacing the existing Pattern management page. The Knowledge management page enables you to better understand and use the Knowledge updates and patterns installed on BMC Atrium Discovery. You can access the Knowledge management page through the Discovery tab.

## Key terminology

- **Pattern** — A sequence of commands written in the Pattern Language (TPL), which contain instructions that identify scanned entities which are then used to create the BMC Atrium Discovery data model.
- **Module** — A text file written in TPL which contains one or more patterns.
- **Knowledge uploads** — Typically these are TKU zip files that include:
  - Zip archives of patterns
  - Hardware Reference Data (HRD) files
  - Device RPMs (DRPMS)

## To view the Knowledge management page

To view the Knowledge management page, from the Discovery tab, click Knowledge Management.

It provides:

- **Toolbar** — a toolbar enabling you to create patterns, upload knowledge, and download pattern modules, view pattern configurations, and enable automatic knowledge cleanup.
- **Knowledge summary** — a summary of the TKUs and Custom Modules, and CAM Modules installed on the BMC Atrium Discovery machine. Shows the total number of active and inactive patterns. Also shows the TPL version in use.
- **Pattern modules list** — a tree view of all pattern modules installed on the BMC Atrium Discovery machine, sorted by product category. From here you can view details and the TPL code of pattern modules.
Product categories

The pattern modules are organized in a browsable hierarchy based on the type of product the modules discover, for example, **Structured Data Management Software > Relational Database Management Systems > Oracle RDBMS**. When writing patterns, you can add `tree_path` metadata to control where the pattern module appears in the hierarchy.

The following non-product related categories are also provided:

- **CAM** — a category for modules created using CAM
- **Unclassified modules** — modules that do not have a specified category, for example, pre-version 10 patterns present on an upgraded system, or user written patterns that do not specify a `tree_path`.

Choosing the pattern modules to display

You can view pattern modules in the main tree view section of the Knowledge management page. Selection tools enable you to choose the pattern modules to display.

To choose the type of pattern module to display using the pattern module name:

1. Enter text to search the pattern module name or `tree path` information in the Pattern modules list.
2. Click **Apply** to view the chosen pattern modules.

You can also choose pattern modules using their source or activation state:

1. Select any or all of the following check boxes:
   - **TKU** — to view TKU supplied pattern modules.
   - **Custom** — to view pattern modules written for your organization.
   - **Generated** — to view pattern modules produced using the CAM or Generate Software Instance Pattern tools.
2. Choose to view pattern modules to display depending on their Active or Inactive state.
   Choose one of the following from the **Display** drop down list:
   - active and inactive
   - only active
   - only inactive
3. Click **Apply** to view the chosen pattern modules.
Selecting pattern modules
You can perform actions on selected pattern modules. The selection options are summarized here:

- To expand a product category branch and show all pattern modules, click the plus icon to the left of the product category.
- To select all pattern modules in a product category branch, click the selection check box to the left of the product category.
- To select individual pattern modules from a product category branch, click the selection check box to the left of the pattern module name.
  The selection options are shown in the Knowledge management screen capture.

The action options are summarized here:

- To activate selected pattern modules, select **Activate** from the **Actions** menu. You are requested for confirmation.
- To deactivate selected pattern modules, select **Deactivate** from the **Actions** menu. You are requested for confirmation.
- To delete selected pattern modules, select **Delete** from the **Actions** menu. You are requested for confirmation.
- To download selected pattern modules, select **Download Selected Modules** from the **Actions** menu. You are requested for confirmation.
- To download all visible pattern modules, select **Download Visible Modules** from the **Actions** menu. You are requested for confirmation.

⚠️ **Note**

Activating a pattern means that it is loaded and can be used to identify software instances. You cannot activate a pattern with a lower version number than one that has previously been activated. Delete the pattern with the higher version number first. All patterns specify the TPL version number that they use. If a pattern uses TPL language features which are from a later TPL version than specified, you cannot activate that pattern.

Viewing and editing a pattern module
You can view details and the TPL code of a pattern module from the **Knowledge Management** page.
To view a pattern module

1. From the **Discovery** tab, click **Knowledge Management**.
   The pattern modules are organized in a browsable hierarchy based on the type of product
   the modules discover.
2. Select the pattern module that you want to view.
   - To expand a product category branch and show all pattern modules, click the plus
     icon to the left of the product category.
   - To close an open product category branch, click the minus icon to the left of the
     product category.
3. Select the Pattern Module that you want to view by clicking on its name.
   The **Pattern Module** page is displayed.

4. From this page you can edit the pattern source or configuration if necessary.
   - Editing the pattern is described in To edit the TPL code of a pattern.
   - Editing the pattern configuration is described in pattern configuration.

To edit the TPL code of a pattern

1. To edit the TPL code of a pattern, from the **Pattern Module** page, click **Edit Module**.

   The page is redisplayed with the pattern source in an editor.
2. Edit the code as required.
3. Click **Apply** to save the changes, or **Cancel** to discard them.
Creating a pattern

You can create a pattern from a pattern template or from CAM, both of which can be accessed from the Knowledge Management page. Additionally, you can create Software Instances (SIs) from discovered processes or discovered services.

You can also use a text editor and then upload the pattern file using the Knowledge Management page.

To create a pattern using a pattern template

1. From the Knowledge Management page toolbar, click **Create Pattern**. The Create Pattern page displays. The Create Pattern page provides a table with each row corresponding to a pattern template. For each pattern, the following information or link displays:
   - **Name** — The name of the pattern. Click a pattern name to view the pattern in the View Pattern template window. The View Pattern Template Window shows a read-only version of the pattern with syntax highlighting where TPL keywords and variables are displayed in different colors to the other text. The Turn Line Numbering On/Turn Line Numbering Off button toggles line numbering and can be used to ease copying and pasting text from one of the templates. A button to close the window is also provided.
   - **Description** — A read-only description of the pattern.
   - **Options** — A download link. Clicking this enables you to download a copy of the pattern to your local file system.

2. Select the pattern that most closely matches the task that you are working on and either copy and paste it into a text editor, or download the pattern using the download link.

3. When you have written the pattern, you can upload it using the **Uploading knowledge** link on the Knowledge Management page. For pattern testing you might need to be able to clean up the data your pattern models. See **Destroying data** for more information.

To create a BAI using CAM

The Create Pattern page provides a link to the CAM documentation. A link is also provided to the **Manual grouping** page which is the start point for the CAM process.

To create a SI pattern from discovered processes

Start with viewing discovered processes and follow guidance provided in creating SIs from discovered processes.
Pattern templates

BMC Atrium Discovery includes patterns that model commonly deployed software. Through monthly Technology Knowledge Updates (TKUs), the Technology Knowledge Network team updates and creates new patterns to increase coverage of the leading products running in global 2000 data centers. Additionally, many template patterns are supplied with BMC Atrium Discovery. They are available from the Create Pattern page.

The following patterns are included:

- **template_simple_si**: Maintain a simple Software Instance based on identifying a process.
- **template_si_instances**: Maintain an identifiable Software Instance triggered on a process with its identity in its command line.
- **template_si_version_path**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found in the path.
- **template_si_version_path_table**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found in the path and looked up in a table.
- **template_si_version_command**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found by running a command.
- **template_si_version_registry**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found in the Windows registry.
- **template_si_version_xml_file**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found in an XML file.
- **template_si_version_package**: Maintain a grouped Software Instance triggered on a process, with the Software Instance version found in package information.
- **template_si_collect_children**: Maintain a Software Instance based on identifying a process, which collects the process' children into a single SI.
- **template_bai_search**: Maintain a Business Application Instance based on identifying a Software Instance and searching for some other components.
- **template_bai_add**: Maintain a Business Application Instance based on identifying each of several Software Instances that contribute to it.
- **template_host_location**: Maintain relationships linking Hosts to Locations based on hostname.
- **template_sql_asset_integration**: Use business contextual information stored in a SQL database to annotate Hosts with ownership and location information.
- **template_sql_deep_discovery**: Issue SQL queries to a discovered database to extract further information about it or other software.
- **template_mainframe_mq**: Maintain an identifiable Mainframe MQDetail and relationships to the parent Message Server Software Instance.
- **template_mainframe_storage**: Maintain an identifiable item of Mainframe Storage based on a TapeDrive or DASDDrive and relationships to the containing StorageCollection and MFParts.
• **template_mainframe_transaction**: Maintain an identifiable Mainframe Transaction and relationships to the parent Transaction Server Software Instance.

• **template_cmdb_cs_augment**: Augment BMC_ComputerSystem CIs synchronized to the CMDB with additional information taken from the BMC Atrium Discovery Host node.

• **template_cmdb_hostname_override**: Override the HostName attribute of BMC_ComputerSystem in the CMDB so it takes only the first component of compound dot-separated hostnames.

• **template_cmdb_location**: Relate BMC_ComputerSystem CIs to BMC_PhysicalLocation CIs in the CMDB.

• **template_cmdb_ss_augment**: Augment BMC_SoftwareServer CIs synchronized to the CMDB with additional attributes taken from the BMC Atrium Discovery Software Instance node.

• **template_cmdb_ss_cti**: Override the default Category, Type, and Item values for BMC_SoftwareServer CIs mapped from particular BMC Atrium Discovery Software Instance nodes.

**Destroying data**

Although it is possible to remove an object from the UI, you should note that the object is actually retained in the datastore. Details can still be displayed for audit purposes, and you can choose to include destroyed objects in searches.

**When would I destroy data?**

In general you should not need to destroy data in production. When testing, you might need to destroy data, such as when developing patterns.

**To destroy an object**

1. Display the View Object page of the relevant object by clicking on a highlighted object on any List page or any Report page. All of the object's current attributes and relationships are listed.
2. From the Actions drop down menu, click **Destroy**. The following message is displayed:
   
   Are you sure you want to destroy this node?

3. To destroy the object, click **OK**.
4. The View Object page is redisplayed, with "This node has been destroyed" at the top of the page. You can select History from the Actions menu to display details of the changes made to the object. The object will not appear in subsequent list pages.

**Note**

You cannot undelete an object that has been destroyed.
Uploading knowledge

You can upload knowledge files to a standalone appliance or a cluster from the Knowledge management page. When uploading to a cluster, you only need to do this once; the knowledge is distributed to all cluster members.

To upload knowledge

Before uploading a TKU file, you must stop the discovery service. This is necessary to install new network device definitions.

Changes in the pattern configuration block revision number may reset the pattern configuration changes to the default values

If you install a Technology Knowledge Update (TKU) which contains an update to a pattern configuration block that requires its major revision number to be incremented, any end-user configuration changes to that pattern (made through the pattern configuration block) are lost and the default pattern values are restored.

To upload knowledge, from the Knowledge management page toolbar:

1. Click Upload.
   The Upload Knowledge dialog displays.
2. Click Browse to locate the file to upload. Select the file from the file browser.
3. Click Upload. The upload process begins and the Upload Knowledge dialog displays the progress of the upload and activation.

When complete, a Knowledge Upload Complete dialog displays showing the results.

A successful knowledge update message.

Downloading pattern modules

You can download a zip archive of all pattern modules from the Knowledge management page. Click Download All and save the zip file.

Configuring patterns in the UI

Patterns can have configuration blocks defined in the pattern source. This provides a number of items that can be configured through the user interface. For information on writing patterns with configuration blocks, see Pattern Configuration.
Changes in the pattern configuration block revision number may reset the pattern configuration changes to the default values

If you install a Technology Knowledge Update (TKU) or manually update a pattern configuration block that requires its major revision number to be incremented, any end-user configuration changes (made through the pattern configuration block) are lost and the default pattern values are restored.

The All Pattern Module Configurations page lists all pattern modules that have configuration blocks defined.

To edit pattern configuration

To edit pattern configuration, from the Knowledge management page toolbar:

1. Click View All Pattern Configurations.
   The All Pattern Module Configurations page displays.
2. For the pattern configuration that you want to edit, click the plus icon to the left of the pattern name. This displays a read only view of configurable items.
3. To make changes to these, click Edit Configuration.
4. Make the changes to the editable fields and click Apply.

Executing patterns manually

Patterns are generally triggered by specific events or changes that occur during a discovery run. Sometimes you might want to run a pattern outside a discovery run (for example, you might be developing patterns against already scanned hosts). To do this you can run a pattern against the nodes contained in a Group.
When you run a pattern manually, it is not the same as triggering it as a result of scanning. Discovery calls can be made from the pattern, but it does not undertake full discovery. If you have changed your discovery credentials or configuration, you should rescan before running patterns manually.

**To select hosts or other nodes**

You can select hosts or other nodes by adding them to a group. From a view node (including host) page, select Groups from the Actions list and add the node to a group. Or, from a report or other search result, select the required items. Then, select Groups from the Actions list and add the nodes to a group.

**Node types against which patterns can be run**

You should add nodes to your group of the kind that the pattern triggers on (for example, if the pattern triggers on a DiscoveredProcess, then you should add DiscoveredProcess nodes). However, the system is able to expand host nodes in an intelligent fashion such that it is possible, for example, to simply add a host even though the pattern requires a DiscoveredProcess.

The following table defines the complete set of traversals used to expand from host nodes to other node kinds. Where more than one traversal is shown, the traversal steps are followed one after the other.

<table>
<thead>
<tr>
<th>Required</th>
<th>Traversal(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered Process</td>
<td>• InferredElement:Inference: Primary:DeviceInfo</td>
</tr>
<tr>
<td></td>
<td>• DiscoveryResult:DiscoveryAccessResult: DiscoveryAccess: DiscoveryAccess:</td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccess</td>
</tr>
<tr>
<td></td>
<td>• DiscoveryAccess:DiscoveryAccessResult: DiscoveryResult:ProcessList</td>
</tr>
<tr>
<td></td>
<td>• List:List:Member:DiscoveredProcess</td>
</tr>
<tr>
<td>Discovered Listening Port</td>
<td>• InferredElement:Inference: Primary:DeviceInfo</td>
</tr>
<tr>
<td></td>
<td>• DiscoveryResult:DiscoveryAccessResult: DiscoveryAccess: DiscoveryAccess:</td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccess</td>
</tr>
<tr>
<td></td>
<td>• DiscoveryAccess:DiscoveryAccessResult: DiscoveryResult:NetworkConnectionList</td>
</tr>
<tr>
<td></td>
<td>• List:List:Member:DiscoveredListeningPort</td>
</tr>
<tr>
<td>Software Instance</td>
<td>• Host:HostedSoftware: RunningSoftware:SoftwareInstance</td>
</tr>
<tr>
<td>Business Application Instance</td>
<td>• Host:HostedSoftware: RunningSoftware:BusinessApplicationInstance</td>
</tr>
<tr>
<td>Device Info</td>
<td>• InferredElement:Inference: Primary:DeviceInfo</td>
</tr>
<tr>
<td>Host Info</td>
<td>• InferredElement:Inference: Primary:HostInfo</td>
</tr>
</tbody>
</table>
To run a pattern

1. From the **Discovery** tab, click **Knowledge Management**.
2. Select the Pattern Module containing the pattern that you want to run.
3. From this page you can edit the pattern source or configuration if necessary. Editing the pattern is described in **Viewing and editing a pattern module**.
   After the pattern is edited, the **Knowledge Management** page displays showing a Save Pattern Module Edit - Complete panel.
4. Select the Pattern Module you want to run from the Pattern Module list.
   The Pattern Module page is displayed.
5. Click the Pattern link in the heading table.
6. From the **Actions** list, select **Run Pattern**.
7. Select the Group that you want to run the pattern against using the Run against Group list. Then choose the settings for the pattern run. Set Expand, Execution Logging, and Additional Discovery. The settings are described in the following table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run against Group</td>
<td>Provides the list to select the group to run the pattern against. If you do not have any Working Sets then the check box for showing only Working Sets will be disabled. If you do have at least one working set then clearing this check box enables you to choose Groups that are not in your working set. The text beneath shows the number of nodes in the group that are the correct node kind to match the pattern's trigger. If the group contains a host node, select the Expand check box to check the host for additional nodes that match the pattern's trigger. For example, the <strong>ApacheBasedWebserver</strong> pattern triggers on <strong>DiscoveredProcess</strong> nodes. If the group contains one <strong>DiscoveredProcess</strong> node and one host node (containing, 162 <strong>DiscoveredProcess</strong> nodes) this field shows 1 <strong>Discovered Process</strong> node if <strong>Expand is not checked</strong> and 163 <strong>Discovered Process</strong> nodes (including 162 via 1 Host node) if it is checked.</td>
</tr>
<tr>
<td>Execution Logging</td>
<td>Select the logging level for this pattern run. This is one of Debug, Info, Warning, Error, or Critical.</td>
</tr>
<tr>
<td>Additional Discovery</td>
<td>Choose whether discovery commands that perform additional discovery should perform live discovery of the host. For example, the <strong>runCommand</strong> method performs additional discovery by calling remote commands from patterns. &lt;br&gt;• <strong>Do not get extra data</strong> — Use any existing data that is available on the appliance. &lt;br&gt;• <strong>Get data as needed</strong> — Use any existing data that is available on the appliance. If additional data is required, perform discovery on the target to obtain it. Get data as needed will only make a request if that request has not been made before.&lt;br&gt;• <strong>Get all new discovery data</strong> — Always perform a new discovery. Do not use any previously discovered data.</td>
</tr>
</tbody>
</table>

While the pattern is running, the results page is displayed.
Configuring pattern cleanup

The data model used in BMC Atrium Discovery version 10 shares pattern module nodes between knowledge upload nodes if the corresponding pattern versions have not changed between upload versions. BMC Atrium Discovery version 9 duplicated the pattern module nodes. Additionally, the components in TKU releases, are now combined into a single upload (formerly known as a pattern package). This work has made it possible to provide automatic cleanup capability to the installed patterns.

Pattern cleanup determines whether old knowledge uploads or pattern modules are obsolete and enables you to remove obsolete knowledge uploads or pattern modules automatically.

When writing TPL patterns it is recommended that you do not modify TKU patterns. If you have modified TKU patterns then auto cleanup will eventually remove them from the system as later versions of the pattern are introduced by subsequent TKUs. When writing a pattern to override a TKU pattern, you should use the `overrides` statement in the pattern declaration.

To cleanup patterns and knowledge uploads automatically

From the Knowledge Management page toolbar, select **Auto Cleanup**. Confirmation is requested.

Enabling auto cleanup performs an initial evaluation of all uploads to see whether any old uploads and pattern modules can be removed. Thereafter, automatic cleanup will be performed as a result of further uploads and actions on pattern modules.

Automatic cleanup can remove pattern modules belonging to a knowledge upload, without removing the upload itself. The upload is only cleaned up once all of its component modules have been removed.

Importing data

This section describes how to import data into BMC Atrium Discovery.

- Importing network device data
- Importing Hardware Reference Data
- Importing CSV data

Importing network device data

The CiscoWorks import tool creates network device entries in the datastore. The import tool creates and updates Network Device nodes.

Entries in the datastore for a network device are overwritten if a network device in the input data has the same name.
A limitation of the importer is that a network device already in the datastore for which a matching name cannot be found in the input data will not be removed. Therefore if a network device has been renamed since a previous import, it will appear in the datastore as a second network device.

⚠️ Use of imported data and direct discovery is not supported

If you use the CiscoWorks import tool to import network devices, and also scan the same network devices, this will potentially generate duplicate network devices. BMC recommends that you use either direct discovery or import, not both.

Importing data from CiscoWorks

The CiscoWorks importer loads the data file exported from CiscoWorks LMS using the Campus Manager User Tracking tool, and loads any existing network device data from the BMC Atrium Discovery datastore. It then compares the sets of data. The supported versions are Campus Manager 3.x and 4.x (UT CLI 1.0, 1.1, and 1.1.1).

The network device name is used as the index. Therefore, if two network devices have the same name, they are treated as the same network device.

- If a new network device is found in the exported file, it is added to the BMC Atrium Discovery datastore. The name of the network device is used as its primary key. See the note below regarding deletion of network devices.
- The ports on each network device are updated. These are indexed by IP address (the primary key). Port attributes that are updated include speed, duplex, and IP address. New ports (identified by IP address) are added, and any ports removed from the CiscoWorks data are removed from the datastore.

Importing a CiscoWorks data file

To import a CiscoWorks Data File:

1. From the Discovery section of the Administration tab, select CiscoWorks Import. The options on this page are described below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>File To Upload</td>
<td>The file to upload. Click Browse... to display a file browser window. Select the file to import and click Open.</td>
</tr>
<tr>
<td>File Type</td>
<td>Choose the type of file to import. Choose either XML or delimited from the File Type list. If you select XML, the lists for File Delimiter and File Has Header Line are dimmed.</td>
</tr>
<tr>
<td>File Delimiter</td>
<td>For a delimited file only, select the delimiter type from the File Delimiter list. This can be one of Comma, Tab, or Space.</td>
</tr>
<tr>
<td>File Has Header Line</td>
<td>Choose Yes or No from the File Has Header Line list depending on whether the delimited file has a heading line.</td>
</tr>
</tbody>
</table>
1. **Remove existing Network Devices**

   Do you want to remove existing network devices from the BMC Atrium Discovery datastore that are not present in the file you are importing? Choose Yes or No from the Remove existing Network Devices list.

2. **Logging Level**

   Choose the logging level from the Logging Level list. This can be Debug, Info, Warning, Error, or Critical.

To import the CiscoWorks data file, click **Apply**.

There is also a command line utility available for importing CiscoWorks data. It is described in `tw_imp_ciscoworks`.

### Generating CiscoWorks data

To export data from CiscoWorks in a layout which the importer can use, you must create a custom layout. To do this:

1. From the CiscoWorks LMS Portal, select **Reports**.
2. Select **Custom Layouts**.
3. Click **Create**.
4. In the Add Layout dialog box, select **End Host**.
5. Select the columns that you want displayed in the report from the Available Sources list and click **Add** to add them to the Selected Sources list.
6. Arrange these into the required order using the up and down arrows.

   The import file must be in the following order otherwise data will be imported into the wrong attribute, and the import will fail. More or fewer columns will also cause the import to fail:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserName</td>
<td></td>
</tr>
<tr>
<td>MACAddress</td>
<td></td>
</tr>
<tr>
<td>HostName</td>
<td></td>
</tr>
<tr>
<td>IPAddress</td>
<td></td>
</tr>
<tr>
<td>Subnet</td>
<td></td>
</tr>
<tr>
<td>DeviceName</td>
<td></td>
</tr>
<tr>
<td>Device</td>
<td></td>
</tr>
<tr>
<td>Port</td>
<td></td>
</tr>
<tr>
<td>PortName</td>
<td></td>
</tr>
<tr>
<td>PortState</td>
<td></td>
</tr>
<tr>
<td>VTPDomain</td>
<td></td>
</tr>
<tr>
<td>VLAN</td>
<td></td>
</tr>
<tr>
<td>VLANType</td>
<td></td>
</tr>
<tr>
<td>LastSeen</td>
<td></td>
</tr>
<tr>
<td>Notes</td>
<td></td>
</tr>
<tr>
<td>PortDuplex</td>
<td></td>
</tr>
<tr>
<td>PortSpeed</td>
<td></td>
</tr>
<tr>
<td>VLANId</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserName</td>
<td>User Name</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC Address</td>
</tr>
<tr>
<td>HostName</td>
<td>Host Name</td>
</tr>
<tr>
<td>IPAddress</td>
<td>IP Address</td>
</tr>
<tr>
<td>Subnet</td>
<td>Subnet</td>
</tr>
<tr>
<td>DeviceName</td>
<td>Device Name</td>
</tr>
<tr>
<td>Device</td>
<td>Device</td>
</tr>
<tr>
<td>Port</td>
<td>Port</td>
</tr>
<tr>
<td>PortName</td>
<td>Port Name</td>
</tr>
<tr>
<td>PortState</td>
<td>Port State</td>
</tr>
<tr>
<td>VTPDomain</td>
<td>VTP Domain</td>
</tr>
<tr>
<td>VLAN</td>
<td>VLAN</td>
</tr>
<tr>
<td>VLANType</td>
<td>VLAN Type</td>
</tr>
<tr>
<td>LastSeen</td>
<td>Last Seen</td>
</tr>
<tr>
<td>Notes</td>
<td>Prefix</td>
</tr>
<tr>
<td>PortDuplex</td>
<td>Port Duplex</td>
</tr>
<tr>
<td>PortSpeed</td>
<td>Port Speed</td>
</tr>
<tr>
<td>VLANId</td>
<td>VLAN Id</td>
</tr>
</tbody>
</table>

a. Specify the name of the layout as **StandardTideway** and enter a description.

b. To create the **StandardTideway** layout, click **Add**.
The full instructions are available on the Cisco Systems website.

The following methods of generating the CiscoWorks data are supported.

### Generating CSV files

Generating CSV files on the CiscoWorks server:

```
C:\Program Files\CSCOpx\campus\bin\ut -cli -query all
   -layout StandardTideway -export c:\data.csv
   -u user -p password
```

This command produces a Java stack trace. This is a known issue and can be ignored. The file that is produced can be imported by running the following command on the BMC Atrium Discovery appliance:

```
$TIDEWAY/bin/tw_imp_ciscoworks --delimiter=","
   --username name --password password
   --file ~/tmp/data.csv
```

### Generating XML import files

Generating XML files on the CiscoWorks server:

```
C:\Program Files\CSCOpx\campus\bin\cmexport ut -u userid
   -p password -host -layout StandardTideway
```

The file is written into the following directory:

```
C:\PROGRAM FILES\CSCOpx\files\cmexport\ut
```

The file that is produced can be imported with the following command on the BMC Atrium Discovery appliance:

```
$TIDEWAY/bin/tw_imp_ciscoworks --xml --username name
   --password password --file ~/tmp/2006516154526ut.xml
```

The following attributes are imported into the data store, where they exist in the input file:
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Example</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MacAddress</td>
<td>MAC address of port on connected device. This is not the MAC address of the network device</td>
<td>01-23-45-67-89-ab</td>
<td>Must be a valid MAC address</td>
</tr>
<tr>
<td>IPAddress</td>
<td>IP address of port on connected device. This is not the IP address of the network device</td>
<td>192.168.0.1</td>
<td>Must be a valid IP address</td>
</tr>
<tr>
<td>DeviceName</td>
<td>Name of network device</td>
<td>switch1.tideway.com</td>
<td></td>
</tr>
<tr>
<td>Device</td>
<td>IP address of network device</td>
<td>192.168.0.1</td>
<td>Must be a valid IP address</td>
</tr>
<tr>
<td>Port</td>
<td>Port name</td>
<td>3/19</td>
<td></td>
</tr>
<tr>
<td>PortDuplex</td>
<td>Duplex setting of port</td>
<td>full-duplex</td>
<td></td>
</tr>
<tr>
<td>PortSpeed</td>
<td>Speed setting of port</td>
<td>100M</td>
<td>Must be a power of ten with K/M/G suffix</td>
</tr>
<tr>
<td>PortNegotiation</td>
<td>Negotiation setting of the Port.</td>
<td>AUTO</td>
<td>Must be AUTO or FORCED</td>
</tr>
</tbody>
</table>

### Importing Hardware Reference Data

Hardware Reference Data (HRD) provides the following vendor specification data:

- Rack Unit size
- Power W
- Power VA
- Heat Output BTU

Import of HRD into BMC Atrium Discovery will enable customers to associate discovered systems with physical assets and report on both directly from BMC Atrium Discovery.

### Importing a Hardware Reference Data file

To import a HRD file:

1. From the Model section of the Administration tab, click **HRD Import**.

   The options on the Import Hardware Reference Data page are described in the following table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>File To Upload</td>
<td>The file to upload. Click <strong>Browse</strong>... to display a file browser window. Select the file to import and click <strong>Open</strong>. See <strong>File Format</strong> for details on the format of HRD files.</td>
</tr>
<tr>
<td>Action</td>
<td>Do you want update, create or delete existing HRD?</td>
</tr>
<tr>
<td></td>
<td><strong>Create and Update</strong> will perform both actions as described below.</td>
</tr>
<tr>
<td></td>
<td><strong>Create</strong> will import records that do not already exist, but will not change any records already imported.</td>
</tr>
<tr>
<td></td>
<td><strong>Update</strong> is used ONLY when non-key fields in an existing record need to be updated. Key fields are <strong>vendor</strong>, <strong>model</strong>, and <strong>config_num_processors</strong>. <strong>Update</strong> will find all imported records that match existing data and update with the imported values. New records in the imported data will <strong>NOT</strong> be created. If you need to</td>
</tr>
</tbody>
</table>
1. Change a key field, you must first delete the existing records and then create the new records. Delete is used to delete records selectively. All records imported that match an existing record will cause that existing record to be removed. Existing records that do not appear in the import set will not be deleted. The HRD patterns automatically recreate the HRD nodes; you do not have to rescan the hosts. Choose Create And Update, Create, Update or Delete from the list.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>change a key field, you must first delete the existing records and then create the new records. Delete is used to delete records selectively. All records imported that match an existing record will cause that existing record to be removed. Existing records that do not appear in the import set will not be deleted. The HRD patterns automatically recreate the HRD nodes; you do not have to rescan the hosts. Choose Create And Update, Create, Update or Delete from the list.</td>
</tr>
</tbody>
</table>

| Logging Level | Choose the logging level from the list. This can be Debug, Info, Warning, Error, or Critical. |

2. Click **Apply**.

**File format**

You can import **csv** files which must have a heading row specifying the HRD node attributes. For example:

```plaintext
vendor,model,known_model_names,config_num_processors,processor_type,max_ram,u_size,power_v,Sun Microsystems,Sun Fire V440,['Sun Fire V440','SUNW,Sun Fire V440'],4,UltraSPARC-IIIi,Sun Microsystems,Sun Fire 280R,['Sun Fire 280R'],2,UltraSPARC-III+,4096,4,454,318,1086,114
```

**Importing CSV data**

It is often very easy to extract information from applications and devices and store it as CSV data. Data can be exported from BMC Atrium Discovery in this format. For more information, see **CSV API**. You can also import CSV data into BMC Atrium Discovery (for example, you might have information on a number of hosts that you want to import).

You can import a CSV file using the **Import CSV Data** page described in the following section. You can choose delimiters other than commas when importing data using the **Import CSV Data** page. Alternatively, if you have command line access, you can use the **tw_imp_csv** utility.

**Incorrect usage may result in data loss**

Before using the **Import CSV Data** page you should fully understand the **system taxonomy** and the changes that you are going to make to your data. Using **Import CSV Data** incorrectly can cause irreparable damage to your data. The data you submit using this tool is applied directly to the production data **without any validation**.

*Always* back up your datastore before using this tool.

**Do not import the following node kinds**
To import a CSV file

1. From the Model section of the Administration tab, click CSV Import.

   The following screen illustrates the Import CSV Data page with the Host Container node kind selected.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>File To Upload</td>
<td>The file to upload. Click Browse to display a file browser window. Select the file to import and click Open.</td>
</tr>
<tr>
<td>Update Mode</td>
<td>Select the mode to use for updates. Choose one of the following from the list:</td>
</tr>
<tr>
<td>Node Kind</td>
<td>Choose the type of node to import from the list. This can be one of any node kind defined in the taxonomy. Changing the node kind updates the Keys field dynamically.</td>
</tr>
</tbody>
</table>
   | Keys | Choose the key or keys for each entry. You must select at least one key. The taxonomy defined keys are marked with a red asterisk *. Use the check boxes if you want to use other keys. The following quick selection links are provided at the top of the keys field:
   |   | Key Attributes: selects just the taxonomy defined keys. |
   |   | None: clears any selection.  
   |   | *Note:* The behavior differs from the tw_imp_csv utility, which uses the first column of the input file as the key unless others are specified. When controlling an import using the UI Import page, only the keys that are selected are used. |
   | Delimiter | Choose the required delimiter from the list. This can be Comma, Semicolon, or Tab. |
   | Logging Level | Choose the logging level from the list. This can be Debug, Info, Warning, Error, or Critical. |

2. To import the CSV file, click Apply.

   See CSV import examples for example of using the CSV importer.

- You must never import DDD nodes.
- You should avoid importing Host nodes and other system maintained nodes. If in doubt, contact Customer Support.
File format

The CSV file represents a two dimensional table with rows separated by newlines, and columns separated by commas. Commas can be included in that data but must be enclosed between double quote characters.

Column names that do not begin with a hash character (#) are treated as names of attributes on the node.

Column names that begin with a hash character are treated as key expressions. Key expressions are used to create or update a relationship to another node. See Relationships for more information.

Keys

The CSV importer searches the datastore for nodes of a specified kind which have a key or keys matching rows in the supplied CSV data. Where the keys match, the node is updated, or deleted and recreated depending on the options selected. The first attribute column is used as a key by default. This can be overridden with the Keys option described in the table above.

Relationships

Relationship columns describe a relationship between the node being modified and one or more other nodes. Relationships are defined using a full key expression. The first example shows a link to a single node:

```
#role:relationship:role:kind.attribute
```

This example shows a link to multiple target nodes. Notice that you will need to surround the relationship definition and data in double quotes so that the subsequent attributes do not get treated as local nodes:

```
"#role:relationship:role:kind.attribute,attribute,attribute"
```

The first role part is the kind of the role that the node being created or updated plays in the relationship. The relationship part is the kind of the relationship being created. The second role part is the kind of role that the other node plays in the relationship. The kind part is the node kind of the other node. Finally, the attribute or attribute list is the name of the attribute used to find the other node. See Node lifecycle for information on relationships.

Only a single traversal is supported.
Type conversion

By default, the CSV importer attempts to convert attribute values to the type defined in the taxonomy. If an attribute is not defined, it is left as a string. If a conversion fails, it is also left as a string.

For list types, the value in the CSV file must be in Python list syntax.

When the `tw_imp_csv` utility is used with the `--force` option, all attributes are left as strings.

Example

To free up rack space for other applications, some hosts have been moved from the Campus data centre to the newly acquired Firehouse data centre. Discovery and Reasoning have handled the IP address and subnet changes but the Host nodes are still linked to the wrong location. Here is the CSV file to process, called `firehouse_move.csv`:

```
name,#ElementInLocation:Location:Location:Location.name
egon,Firehouse
ray,Firehouse
peter,Firehouse
```

We process the CSV file with the following command line:

```
$ tw_imp_csv --username=system --password=system --kind=Host firehouse_move.csv
```

The script reads the file called `firehouse_move.csv` line by line. It uses the first line to name the columns. The first column is called `name`, which does not begin with a `#` character so it is treated as an attribute name. The second column does begin with a `#` character so it is treated as a specification for some relationships.

No explicit key has been specified so the first (and in this case, only) attribute name is taken to be the key.

Now the script reads the second line. The first field, `egon`, is in the `name` column which was selected as the key earlier. So the script uses the search service to find a node of kind Host (from the `--kind` command line option) that has a `name` attribute equal to `egon`. It finds exactly one node matching that search. If it had not found that node, it would have been created. If it had found multiple nodes, an error would have been reported and processing would continue with the next line, NO nodes would have been updated.
Having found the node, it updates it using the other fields on the row. Were there any other attribute columns in the file, the script would have used these to update the node before looking at the relationships.

The file has only one relationship column. The name is `#ElementInLocation:Location:Location:Location.name`. The script searches for a Location node with a `name` attribute equal to Firehouse, this row's value for the column. Having found the Location node, the script creates a `HostLocation` relationship to it with the Host node playing the `HostInLocation` role and the Location playing the `LocationOfHost` role.

The script then processes the second and third data rows, updating the `ray` and `peter` nodes with the new location.

### Managing your IT infrastructure

The Infrastructure section of BMC Atrium Discovery includes details of the system physical model, the hardware and software items in your network. This data is obtained from the systems during the Discovery process.

- Viewing details of the IT infrastructure
- Searching the IT infrastructure
- Viewing a host
- Viewing a cluster
- Viewing a host container
- Viewing a mainframe
- Viewing a network device
- Viewing load balancers
- Viewing a network printer
- Viewing an SNMP managed device
- Viewing a subnet
- Viewing a software instance
- Viewing storage
- Reporting on the infrastructure

For more information about the BMC Atrium Discovery model, see the Data Model.

### Viewing details of the IT infrastructure

On the main page of the Infrastructure tab you can view a summary of the infrastructure items held in the system, the infrastructure items that are owned by you, as well as a number of infrastructure-related reports.

From any list of objects you can drill down to display more detailed information (for example, to view the details of a host or a software instance and the relationships between objects).
To view a summary of infrastructure items

1. Click the **Infrastructure** tab in the primary navigation bar.
   The Infrastructure page lists the total number of each kind of infrastructure item in the system, as well as any infrastructure objects assigned to you. It also displays a list of infrastructure-related reports. See Reporting on the infrastructure.

2. To view a list of infrastructure items, click any of the listed object kinds.
   A list of the appropriate objects of the selected type is displayed.

To view details of an infrastructure item

1. Display an infrastructure list page, for example:
   - A list of your infrastructure objects.
   - The results of any infrastructure search.
   - Any infrastructure report.

2. Click any highlighted object in the list to display the View Object page, which lists the attributes of the object and its relationships with other objects.

From this page you can view any other related object or delete the object.

⚠️ **Note**

Some attributes or relationships only appear on the View Object page if a value is set. If data has not been discovered, these fields are not listed.

Searching the IT infrastructure

You can search for information in the BMC Atrium Discovery datastore by clicking the **Search Options** icon, which displays a menu where you can tailor your searches on all pages in BMC Atrium Discovery.

To run a basic infrastructure search

1. In the Search Options menu, type the keyword or text string you want to search for.
2. Choose **Infrastructure** from the menu.
3. Choose **Exact Match** to search for an entire attribute, or choose **Word Match** to search for an entire word.
4. Select the **Show Data Completeness** check box if you want to view the data completeness issues for all items.
5. Click **Search**.

Summary details of infrastructure objects matching your criteria are listed.

6. If your search returns too many matches, you can refine it by running an additional search on the items.
For more searching options, select **Advanced Search** from the Search box. For more information about searching, see **Searching for data**.

**Viewing a host**

All hosts are set up automatically by BMC Atrium Discovery.

- To view a host
- General Details
- Identity
- Ownership & Location
- Operating System
- Hardware and Network
- Inference
- CMDB Synchronization
- Other
- To run host reports
- Network Interfaces field
- Network interface notes
- File Systems field
- Disk Drives
- Storage volumes

**To view a host**

1. Click the **Hosts** link in the Infrastructure Summary section of the Infrastructure page.
2. Select a host from the list.
   
   The BMC Atrium Discovery Host page groups the field names in a logical format.
This screen illustrates the Host page divided into separate groups of information that you can view by collapsing or expanding.

The information fields for a host object are arranged in the following groups:

- **General Details**: General information on the host such as its name, type, and hosted applications. See [General Details](#).
- **Identity**: Information on FQDN and domain. See [Identity](#) for details.
- **Ownership & Location**: Information on the ownership and location of the host. See [Ownership & Location](#) for details.
- **Operating System**: The OS and service pack of this host. See [Operating System](#) for details.
- **Hardware and Network**: Information on the physical hardware installed on the host, and network information. See [Hardware and Network](#) for details.
- **Inference**: The inference details. See [Inference](#) for details.
- **Other**: See [Other](#) for details.

⚠ **Note**

You can also display provenance information by clicking the Show Provenance button. Provenance information is meta-information describing how the other information came to exist. For more information, see [Provenance Information](#).

### General Details

A list of information fields that can be displayed in the General Details group for a host object are detailed in the following table.

**General information fields for a host object**
<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this host system. A link is also provided which generates a pdf Host Profile. Click this to generate the Host Profile and then download it by clicking on the Reports Tab. See Host Profile for more information.</td>
</tr>
<tr>
<td>Host Type</td>
<td>The type of the host, such as UNIX Server, Windows Desktop, and so on.</td>
</tr>
<tr>
<td>Hardware Vendor</td>
<td>The vendor of this host.</td>
</tr>
<tr>
<td>Partition</td>
<td>Whether the host is running on a partition.</td>
</tr>
<tr>
<td>Communicating With</td>
<td>A link to the host or list of host that this host is communicating with.</td>
</tr>
<tr>
<td>Virtual</td>
<td>Is this host a virtual machine. Only displayed when true.</td>
</tr>
<tr>
<td>Hosted Applications</td>
<td>The hosted applications on this host.</td>
</tr>
<tr>
<td>Software Instances</td>
<td>The Software Instances related to this host. Click the link to view the related Software Instances. If there are storage systems which are managed by the host, those are also displayed. Click the link to view the Storage System List page.</td>
</tr>
<tr>
<td>Aggregate Software Instances</td>
<td>The Aggregate Software Instances related to this host. Click the link to view the related Aggregate Software Instances.</td>
</tr>
<tr>
<td>Runtime Environments</td>
<td>The Runtime Environments related to this host. Click the link to view the related Runtime Environments.</td>
</tr>
<tr>
<td>Virtual Machines</td>
<td>The names of the virtual machines that run on this host. These are created by the VMware pattern.</td>
</tr>
<tr>
<td>Containing Host</td>
<td>The containing host for the virtual machines on this host, that is, the chassis in which it runs.</td>
</tr>
<tr>
<td>Containing VM</td>
<td>The containing virtual machines for this host, that is, the virtualization software and server that this host is on.</td>
</tr>
<tr>
<td>Zonename</td>
<td>Zonename of this host system. This is for Solaris zones only.</td>
</tr>
<tr>
<td>Zone UUID</td>
<td>The zone UUID of this host system. This is for Solaris zones only.</td>
</tr>
<tr>
<td>Host Container</td>
<td>The container for this host, that is, the machine on which it runs.</td>
</tr>
<tr>
<td>Cluster</td>
<td>The name of the cluster. This is only applicable if this host is a member of a cluster of host machines.</td>
</tr>
<tr>
<td>E10K SSP Hostname</td>
<td>The name of the host on which the E10K System Service Processor runs.</td>
</tr>
<tr>
<td>SunFire Domain</td>
<td>The SunFire domain name.</td>
</tr>
<tr>
<td>LDOM</td>
<td>The LDOM in which this host is running.</td>
</tr>
<tr>
<td>Power LPAR Number</td>
<td>the number of the LPAR on which the host is running.</td>
</tr>
<tr>
<td>Files</td>
<td>Configuration files for this host.</td>
</tr>
</tbody>
</table>
Identity

A list of information fields that can be displayed in the Identity group for a host object are described in the following table.

Identity fields for a host object

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hostname Aliases</td>
<td>Other names by which this host is known. List of string values.</td>
</tr>
<tr>
<td>Local FQDN</td>
<td>The fully qualified domain name which is local to the host.</td>
</tr>
<tr>
<td>DNS Domain</td>
<td>The DNS domain that this host is part of.</td>
</tr>
<tr>
<td>NIS/Windows Domain</td>
<td>The name of the NIS or Windows domain that this host is in.</td>
</tr>
<tr>
<td>Windows Workgroup</td>
<td>The Windows workgroup that this host belongs to.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>If the endpoint used to discover this host was previously identified as a different host, a link to that host and a link to a comparison with that host are provided.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>If the endpoint used to discover this host is now identified as a different host, a link to that host and a link to a comparison with that host are provided.</td>
</tr>
</tbody>
</table>

Ownership & Location

A list of information fields that can be displayed in the Ownership & Location group for a host object are described in the following table.

Ownership and location fields for a host object

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizational Unit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>The person or owner responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>The person or owner responsible for this element from an IT perspective.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>The person or owner responsible for the support of this element.</td>
</tr>
<tr>
<td>Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Status</td>
<td>The lifecycle status of this element.</td>
</tr>
</tbody>
</table>

Operating System

A list of information fields that can be displayed in the Operating System group for a host object are described in the following table.

Operating system information fields for a host object

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The discovered OS for this host, for example, Red Hat Enterprise Linux Server release 5 (Tikanga).</td>
</tr>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Discovered OS</td>
<td>The discovered OS class for this host, for example, UNIX.</td>
</tr>
<tr>
<td>Discovered OS Class</td>
<td>The discovered OS type for this host, for example, Red Hat Enterprise Linux.</td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>The discovered OS version for this host, for example, 5.</td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>The discovered OS edition for this host, for example, Server.</td>
</tr>
<tr>
<td>Discovered OS Update Level</td>
<td>The discovered OS update level for this host, for example, 6100-02.</td>
</tr>
<tr>
<td>Discovered OS Architecture</td>
<td>The discovered OS architecture for this host, for example, x86_64.</td>
</tr>
<tr>
<td>Discovered OS Build</td>
<td>The discovered OS build for this host, for example Tikanga.</td>
</tr>
<tr>
<td>Discovered OS Vendor</td>
<td>The discovered OS vendor for this host, for example, Red Hat.</td>
</tr>
<tr>
<td>Service Pack</td>
<td>The service pack installed on this host.</td>
</tr>
<tr>
<td>Discovered Kernel</td>
<td>The discovered kernel for this host.</td>
</tr>
<tr>
<td>Patch Count</td>
<td>The number of software patches discovered on this host.</td>
</tr>
<tr>
<td>Patches</td>
<td>The software patches related to this host.</td>
</tr>
<tr>
<td>Package Count</td>
<td>The number of software packages discovered on this host.</td>
</tr>
<tr>
<td>Unique Packages</td>
<td>The software packages related to this host.</td>
</tr>
<tr>
<td>OS Support Details</td>
<td>Operating System Support details. This provides the dates for the OS End of Life, End of Support, and End of Extended Support. Clicking any of the links displays the Hardware Reference Data page for that host type. For more information, see OS Support Page.</td>
</tr>
</tbody>
</table>

**Hardware and Network**

A list of information fields that can be displayed in the Hardware and Network group for a host object are described in the following table.

⚠️ From version 10.0 onwards, BMC Atrium Discovery includes separate attributes for physical and logical RAM on host nodes.
## Hardware and network information fields for a host object

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Model number of this host.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>Serial number of this host.</td>
</tr>
<tr>
<td>Unique Host Id</td>
<td>ID of this host, found by the discovery system.</td>
</tr>
<tr>
<td>UUID</td>
<td>The UUID of this host.</td>
</tr>
<tr>
<td>Logical RAM</td>
<td>Amount of RAM available on a host, measured in MB.</td>
</tr>
<tr>
<td>Physical RAM</td>
<td>Amount of RAM installed on a host, measured in MB.</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>The number of physical processors.</td>
</tr>
<tr>
<td>Processor Type</td>
<td>The type of processor this host has.</td>
</tr>
<tr>
<td>Number of Logical Processors</td>
<td>The number of logical processors contained on the physical processors that this host has. Each physical processor might have more than a single logical processor. This can be the same as the number of physical or virtual processors.</td>
</tr>
<tr>
<td>Cores per Processor</td>
<td>The number of cores per processor. For example, this could be dual-core or quad-core.</td>
</tr>
<tr>
<td>CPU Threading Enabled</td>
<td>Whether CPU hardware threading is enabled.</td>
</tr>
<tr>
<td>Number of Processor Types</td>
<td>The number of physical processor types.</td>
</tr>
<tr>
<td>Threads per Processor Core</td>
<td>The number of threads per processor core in multi/hyper threaded processors.</td>
</tr>
<tr>
<td>Power Supply Status</td>
<td>BMC Atrium Discovery can identify which hosts have a single PSU, those that have one or more failed PSUs and those for which PSU information is not available. Discovery detects the power supply or power supplies for each host and displays the status of each PSU on the Host page. The status of each PSU is shown as 'OK' or 'FAILED'. If the presence of a PSU is detected, but not its state, that PSU will be shown as 'UNKNOWN'.</td>
</tr>
<tr>
<td>Hardware Reference Data</td>
<td>A table of information on the physical details of this host. For example, the hardware's size in Rack Units, and its power capacity, and thermal characteristics. Clicking any of the links displays the Hardware Reference Data page for that host type. For more information, see Hardware Reference Data Page.</td>
</tr>
<tr>
<td>Networking Interfaces</td>
<td>A detailed view of the network interface or interfaces discovered on this host. See Networking Interfaces field for more information.</td>
</tr>
<tr>
<td>HBA Interfaces</td>
<td>The HBA interface or interfaces discovered on this host. See HBA modeling for more information.</td>
</tr>
</tbody>
</table>
### Field name

- **File System**: A detailed view of the file system for this host. See #File System field for more information.
- **Primary Domain Controller**: The name of the primary domain controller.

### Inference

A list of information fields that can be displayed in the Inference group for a host object are described in the following table.

#### Inference information fields for a host object

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Lists Discovery Accesses for this host, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run of which this is a part, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess Page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.</td>
</tr>
<tr>
<td>Discovery Conditions</td>
<td>Discovery conditions which apply to this host.</td>
</tr>
</tbody>
</table>

### CMDB Synchronization

A list of information fields that can be displayed in the CMDB Synchronization group for a host object are described in the following table.

#### CMDB Synchronization information fields for a host object

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last successful CMDB sync</td>
<td>The time and date at which this host was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time and date at which this host was last unsuccessfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>The time (in seconds) spent performing the last CMDB synchronization for this host.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>The number of CIs corresponding to this host at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>The number of relationships to CIs corresponding to this host at the last CMDB synchronization.</td>
</tr>
</tbody>
</table>

### Other

A list of information fields that can be displayed in the Other group for a host object are described in the following table.

#### Other information fields for a host object
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Additional details of this host.</td>
</tr>
<tr>
<td>Collections</td>
<td>A collection which this Host is a member of.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

⚠️ **Note**

The data in BMC Atrium Discovery is read-only. BMC strongly recommends that you do not edit any nodes because your version will not be retained by BMC Atrium Discovery.

**To run host reports**

A number of context-sensitive reports are available for hosts. See Host Reports for further information and examples of these host-related reports.

**Network Interfaces field**

The **Network Interfaces** field contains information on the host’s network interfaces and their settings. The display of network information has been modified to support IPv6 and complex configurations such as bonded interfaces. For each interface, the following information is shown:

<table>
<thead>
<tr>
<th>Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the network interface which is also a link to the page for that IP address.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address, also a link to the page for that IP address.</td>
</tr>
<tr>
<td>MAC Address</td>
<td>The MAC address of the network interface which is also a link to the page for that network interface.</td>
</tr>
<tr>
<td>Interface</td>
<td>The interface name (eth0, eth1) which is also a link to the page for that network interface. A red x icon is displayed where an interface has some mismatch (speed, duplex, or negotiation) with the network device that it is connected to.</td>
</tr>
<tr>
<td>Connected To</td>
<td>The network device that the network interface is connected to. A red x icon is displayed where the network device port has some mismatch (speed, duplex, or negotiation) with the interface connected to it.</td>
</tr>
</tbody>
</table>

Click **Show Details** to reveal more information on the network interfaces. Where a mismatch has been flagged in the **Interface** or **Connected To** fields, the reason is highlighted as shown in the following screens.

The layout of the blocks indicates the configuration of the interfaces. For example, the following screen shows three IP addresses "spanned" by a single MAC address. In this case, a single interface has an IPv4 address, an IPv6 address, and an IPv6 link local address. Currently, bonded interfaces are correctly displayed for Linux hosts only.
This screen shows the network interfaces field where there is a highlighted mismatch in interface eth0.

This screen shows the same interface after clicking **Show Details** to show additional information and reveal the reason for the highlighted mismatch.

This screen shows the same network interfaces field where **Show Details** has been clicked to show more information on the interfaces and reveal the reason for the highlighted mismatch.

A bonded interface is shown in the following screen where a single MAC address spans three interfaces.

**Network interface notes**

- Hyper-V Windows virtual machines (VMs) always report NIC speed as 10 GBps regardless of the actual speed. Consequently, for Hyper-V Windows VMs, the host and switch mismatch field shows incorrect results.

- In a non-administrator user discovery of a Windows 2003 host using WMI, the Manufacturer attribute of the network interfaces will not be populated.

- The negotiation, speed, duplex, driver_date and driver_version attributes are not set for Windows NT4 systems. The information is not stored in the registry as in later versions of Windows.

**File Systems field**

The **File Systems** field contains information on the local, exported, and remote file systems mounted on the host.
This screen illustrates an example of file system information for a UNIX system.

This screen illustrates an example of file system information for a Windows system.

This screen illustrates an example of file system information for virtual (local) file system on a VMware ESXi 4.0.0 system.

The links in each row link to the File System node page for that file system. The first displays a LOCAL file system node.

The second displays a node EXPORTED from one host linked to a REMOTE node from another.

(For the nfs file system from the client and server point of view).

This screen illustrates an example of a node EXPORTED from one host linked to a REMOTE node from another.

This screen illustrates an example of the first row that links to a LOCAL file system node.

The information fields for a File System node are shown in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>File system name.</td>
</tr>
</tbody>
</table>
### Field name | Details
--- | ---
**Kind** | The kind of file system. This can be LOCAL, EXPORTED, or REMOTE.
**Type** | The type of file system. For example, ext3, tmpfs, NTFS, cifs and so forth.
**Mount** | The file system mount point.
**Size** | The size of the file system.
**Used** | The amount of the file system that is used.
**Used Percentage** | The percentage of the file system that is used. This field and the following fields are highlighted yellow when 81% or more of the capacity is used, and red when 91% or more of the capacity is used. This can be seen in the VMware ESXi 4.0.0 example above.
**Free** | The amount of free space remaining.
**Free Percentage** | The percentage of the file system that is free.
**Host** | The name of the host on which the file system is mounted.
**Disk Drive** | Details on the disk drive holding the file system. May be Physical, Virtual, or SAN.

---

**Note**

BMC Atrium Discovery reports the size and free percentage of the file system. In some file systems, part of the space is reserved for root (or similar) access and is not usable by general users. Thus, the reported size and free space might exceed that available to general users.

**Disk Drives**

This screenshot shows the details on the physical disk drives on the host. This is a VMware ESX host reporting the local physical disk drives on the host.
Storage volumes

The following screenshot shows a physical host linked to the volume on which it is consuming storage.

Consolidated hosts

Consolidated hosts are the discovered hosts on the network that comprise collaborating or virtual hosts that can be grouped into their host containers or clusters. The consolidation of these hosts is managed by standard Pattern Language patterns. This topic discusses types of consolidated hosts supported by BMC Atrium Discovery.

- Consolidation of Starfire domains
- Consolidation of discovered virtual hosts into parent hosts
- Consolidation of cooperating hosts into clusters

Consolidation of Starfire domains

The large Oracle Sun Enterprise servers are modeled as a physical device (the partitioned host) which acts as a container for a number of logical devices (individual servers, as seen on the network). BMC Atrium Discovery discovers the logical devices and pattern modules consolidate these logical devices into physical devices.

Sun Enterprise 10000

The physical device is not directly discovered for Sun Enterprise 10000 servers. However, each logical device has details of its physical server, the SSP host name, which is used to create or update the physical host node representing the physical device in the datastore. The name given to the Sun Enterprise 10000 physical host is the SSP hostname.
Sun Fire servers

The physical device is not directly discovered for Sun Fire servers. However, using the last number of the IP address, the host ID can be decremented to determine the host ID of the system controller. This is used to assign logical devices to physical devices and is also used as the host name of the physical device.

Domain information referred to in the manufacturer’s documentation

<table>
<thead>
<tr>
<th>IP</th>
<th>Example host ID</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.1.1</td>
<td>82a8aaa3</td>
<td>System Controller</td>
</tr>
<tr>
<td>10.1.1.2</td>
<td>82a8aaa4</td>
<td>A</td>
</tr>
<tr>
<td>10.1.1.3</td>
<td>82a8aaa5</td>
<td>B</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>10.1.1.19</td>
<td>82a8aab5</td>
<td>R</td>
</tr>
</tbody>
</table>

Patterns are provided to consolidate the following machines:

- Sun Enterprise 10000 - E10K
- Sun Fire 12000 - F12K
- Sun Fire 15000 - F15K
- Sun Fire 20000 - F20K
- Sun Fire 25000 - F25K

Consolidation of discovered virtual hosts into parent hosts

Where a number of virtual hosts are present on one machine (they are seen as individual IP devices) they are consolidated into a single parent host (also seen as an IP device).

- Solaris 10 Zones
- VMware VMs

This creates a group of SIs, which are linked to the virtual hosts.

Solaris 10 zones

When a Solaris 10 host is discovered, Discovery runs commands to determine the zone name of the host and a software instance is created for all installed zones.

The patterns search for a host with the zone name of global; this is the container host.

The list of installed zones in that host gives the list of zone names to search for on other Solaris 10 hosts. The patterns search for these hosts in the datastore and then link them to the SI for the zone Host node representing the container host. The installed zones are shown as VMs on the container host details page.

⚠️
Note

You cannot run `ls/of` inside a Solaris zone container, by design of the Solaris OS. All processes from all zones are visible in the global zone.

VMware consolidation

The host running the VMware instance is seen as an individual host. When the host is scanned, software instances representing the VMware instance and any running VMware guest OSs are created.

Software Instance nodes are linked to the Host node representing the physical server running the VMware instance and the Host node representing the virtual host.

Consolidation of cooperating hosts into clusters

A Cluster node represents a group of hosts collaborating to form a cluster. The Cluster node represents the group rather than a single physical entity. The types of clusters currently consolidated are:

- Veritas clusters
- Microsoft clusters

These are described in the following sections.

Veritas consolidation rule

The Veritas cluster consolidation pattern triggers on Software Instances of the type VERITAS Cluster Server. Where these have a `cluster_id` attribute defined, which is considered to be the unique cluster identifier, the pattern creates (updates) the Cluster node. It then links each Software Instance with the same `cluster_id` attribute.

Microsoft clusters consolidation rule

When a Windows host is discovered, the following registry keys are read as standard:

- `HKEY_LOCAL_MACHINE\Cluster\ClusterInstanceId` populates the `cluster_instance_id` attribute.
- `HKEY_LOCAL_MACHINE\Cluster\ClusterName` populates the `cluster_name` attribute.
- `HKEY_LOCAL_MACHINE\Cluster\ClusterNameResource` populates the `cluster_name_resource` attribute.

These attributes are part of the default taxonomy; they are created on a Host node when the WMI query returns results. If these queries return results then the host is assumed to be part of a Microsoft cluster.
The Microsoft cluster consolidation pattern triggers on Windows Host nodes with the `cluster_instance_id` attribute, which is considered to be the unique cluster identifier. Hosts with the same value set in this attribute are members of the same cluster. If this is set then the pattern creates (updates) the Cluster node and links to each host with the same `cluster_instance_id` attribute.

### Hardware Reference Data page

The Hardware Reference Data page provides information about the physical and thermal characteristics of the hardware.

You can access the Hardware Reference Data page from the Hardware and Network section of a Host page by clicking any of the links in the Hardware Reference Data section.

#### Information fields for a typical host

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Vendor</td>
<td>The name of the hardware vendor.</td>
</tr>
<tr>
<td>Hardware Model</td>
<td>The hardware model name.</td>
</tr>
<tr>
<td>Known Hardware Model Names</td>
<td>Any other names by which this hardware is known.</td>
</tr>
<tr>
<td>Processor Type</td>
<td>The processor type</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>The number of physical processors in the hardware.</td>
</tr>
<tr>
<td>Maximum RAM</td>
<td>The maximum amount of RAM supported by this hardware.</td>
</tr>
<tr>
<td>Rack Unit (RU)</td>
<td>The size of the hardware.</td>
</tr>
<tr>
<td>Power Capacity (VA)</td>
<td>The input power capacity measured in Volt Amperes.</td>
</tr>
<tr>
<td>Power Capacity (W)</td>
<td>The input power capacity measured in Watts.</td>
</tr>
<tr>
<td>BTU per hour</td>
<td>Thermal output measured in BTUs.</td>
</tr>
<tr>
<td>Power Capacity VA per RU</td>
<td>The input power capacity measured in Volt Amperes, per Rack Unit.</td>
</tr>
<tr>
<td>Power Capacity W per RU</td>
<td>The input power capacity measured in Watts, per Rack Unit.</td>
</tr>
<tr>
<td>BTU per hour per RU</td>
<td>Thermal output measured in BTUs, per Rack Unit.</td>
</tr>
<tr>
<td>Hardware</td>
<td>A link to the host or a list of all hosts using this hardware.</td>
</tr>
</tbody>
</table>

New Hardware Reference data can be imported into BMC Atrium Discovery by an administrator. For more information, see Importing Hardware Reference Data.

### OS Support Detail page

The OS Support Detail page shows the details on the support available from the publisher of the software or OS. The OS Support information is available when the `TKU-EOL-Data` pattern package is uploaded to the appliance and activated. The `TKU-EOL-Data` pattern package is included in the `Extended-Data-Pack-YYYY-MM-N-RELEASE.zip` file. See Knowledge management for more information.
You can access the OS Support page from the Operating System section of a Host page by clicking any of the links in the OS Support Details section.

**Information fields for a typical OS Support Detail page**

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The type of support details page. This is the OS Support Detail Page.</td>
</tr>
<tr>
<td>Software/OS Publisher</td>
<td>The name of the publisher of the software or OS.</td>
</tr>
<tr>
<td>Product</td>
<td>The product (Operating System) name.</td>
</tr>
<tr>
<td>Software/OS Product Version</td>
<td>The version number of the software or OS.</td>
</tr>
<tr>
<td>Software/OS Edition</td>
<td>The edition of the OS. Typical values here are “Professional”, “Server”, or “Home”.</td>
</tr>
<tr>
<td>Retirement (EOL) Date</td>
<td>The End of Life date of the OS. A date is given and a calendar duration until that date.</td>
</tr>
<tr>
<td>End of Support (EOS) Date</td>
<td>The End of Support date of the software or OS. A date is given and a calendar duration until that date.</td>
</tr>
<tr>
<td>End of Extended Support (EOES) Date</td>
<td>The End of Extended Support date of the software or OS. A date is given and a calendar duration until that date.</td>
</tr>
<tr>
<td>Retirement Date defaulted to EOS date</td>
<td>Occasionally a Retirement (EOL) date is not provided for a software product or an OS, when this occurs, the End of Support date is used as the Retirement date. When this is done, this field states True, otherwise False.</td>
</tr>
<tr>
<td>Known Names</td>
<td>Other names that the software or OS is known as.</td>
</tr>
<tr>
<td>Related Hosts (OS)</td>
<td>A link to the host or a list of all hosts running this software product or OS.</td>
</tr>
</tbody>
</table>

**HBA modeling**

Host Bus Adapter (HBA) modeling provides the ability to determine which firmware version is running on individual Fiber Channel HBA cards in the network under Discovery and report on them.

- Supported platforms
- HBA discovery limitations
- To view a list of HBA interfaces

**Supported platforms**

HBA modeling has been extended in BMC Atrium Discovery 8.3 to retrieve information from QLogic cards in addition to Emulex. The methods have been improved so they should be able to retrieve information form other vendors’ cards. These methods are supported on the following platforms:

- Linux (2.6 and 2.4 kernels)
- VMware ESX/ESXi
• Oracle Solaris
• IBM AIX
• Microsoft Windows

In BMC Atrium Discovery versions 9.0 SP1 and later, HBA modelling has been extended for the HP-UX platform and retrieves information from Emulex, QLogic, and HP Tachyon cards.

**HBA discovery limitations**

HBA discovery is not possible on older kernels (pre-Linux kernel 2.4) which do not support the sys pseudo file system. The sys pseudo file system is the only generic technique available.

On VMware ESX/ESXi systems, the proc pseudo filesystem can be used to determine some of the properties of the HBA drivers. Access to proc is only available when discovering VMware ESX /ESXi hosts by using ssh.

On older non-VMware ESX/ESXi kernels, only Emulex HBAs can be discovered because the tools required for discovery are installed along with the driver. Privileged credentials are required to use these tools.

Generally an HBA card cannot be discovered unless:

- There is generic OS support (2.5 and later Linux kernels).
- Vendor tools are installed.

QLogic supply a utility called `scli`, although as an optional installation item it cannot be relied on to be present.

**To view a list of HBA interfaces**

When an HBA card is discovered, the information is displayed on the host page in the user interface. The HBA interface(s) row is divided into fields. The discovered HBA card ID and the firmware version that the card reported are displayed. This identifier string is set by the manufacturer to identify that particular version of the firmware.

The WWNN and WWPN columns display the World Wide Node Number and World Wide Port Number respectively. A Fiber Channel target is assigned its WWNN at loop initialization time. It is possible for the WWNN to change between one loop initialization and the next. Every time the system boots or a target is added to or removed from the Fiber Channel, the loop is re-initialized.

The Connected To column displays the storage systems that the HBA is connected to. It is also a link to the corresponding Fibre Channel HBA page.

There are two possible configurations for HBA cards. You can have an HBA card with one node and two ports or you can have an HBA card with two nodes and one port each. The Emulex cards have two nodes and one port each.
1. Click any of the listed HBA cards. The Fiber Channel HBA page is displayed for that card.
2. From here you can view any other related objects, view history, or destroy the object.

**Viewing a cluster**

A cluster object in BMC Atrium Discovery represents a grouping of one or more logical hosts into a logical group. All cluster objects are set up automatically by the BMC Atrium Discovery Discovery process.

**To view a cluster**

1. Click the **Clusters** link in the Infrastructure Summary section of the Infrastructure page.
2. Select a cluster from the list, as illustrated in the following screen.

![Clusters list](image)

The information fields for a typical cluster are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this cluster.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of cluster, for example, Microsoft cluster.</td>
</tr>
<tr>
<td>Member Hosts</td>
<td>Hosts that are members of this cluster.</td>
</tr>
<tr>
<td>Service Software</td>
<td>Software Instances providing the cluster service.</td>
</tr>
<tr>
<td>Data Quality Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

**Viewing a host container**

A host container object in BMC Atrium Discovery represents a single physical device which is subdivided into multiple logical hosts, typically this would be a blade enclosure. The enclosure is discovered when a management controller is discovered using SNMP. This is modeled as a Management Controller node. The creation of a Management Controller node triggers an additional pattern which creates the Host Container node and collects additional information on the contents of the enclosure, and triggers further patterns to relate the components.

- Some host containers, such as the Sun Blade 6000 system, may need to be scanned twice before all relationships are created.
To view a host container

1. Click the **Host Containers** link in the Infrastructure Summary section of the Infrastructure page.
2. Select a host container from the list, as illustrated in the following screen.

![Host Container Screen](image)

The information fields for a typical host container are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this host container. For example, lon-m100e-02</td>
</tr>
<tr>
<td>Type</td>
<td>The type of host container. For example, Blade Enclosure.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name of this host container. For example, PowerEdge M1000e.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>The serial number of this host container. For example, 417P95J.</td>
</tr>
<tr>
<td>Hardware Vendor</td>
<td>The hardware vendor name of this host container. For example, Dell.</td>
</tr>
<tr>
<td>Host Members</td>
<td>Links to host pages of the hosts contained in the blade enclosure.</td>
</tr>
<tr>
<td>Network Device Members</td>
<td>Links to any network device blades contained in the blade enclosure.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>Links to any SNMP managed devices contained in the blade enclosure.</td>
</tr>
<tr>
<td>Members</td>
<td>Typically this is the management controller.</td>
</tr>
<tr>
<td>Pattern</td>
<td>The pattern used to discover the blade enclosure and its contents.</td>
</tr>
<tr>
<td>Details</td>
<td>Links to pages for the hosts, network devices, and SNMP managed devices contained in the blade enclosure.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

**Viewing a management controller**

A management controller in BMC Atrium Discovery represents a chassis controller on a blade enclosure, or a lights-out management board on a host. The chassis controller/lights-out management board is currently discovered using SNMP.
To view a management controller

1. Click the Management Controllers link in the Infrastructure Summary section of the Infrastructure page.

2. Select a management controller from the list, as illustrated in the following screen.

The information fields for a typical management controller are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Name</td>
<td>Name of this management controller. For example, IDRAC-C1KN95J.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of management controller. For example, Remote Access Controller.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The hardware vendor name of this management controller. For example, Dell.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name of this management controller. For example, iDRAC.</td>
</tr>
<tr>
<td>Testing Status</td>
<td>Whether the exact device has been tested by BMC. This can be either:</td>
</tr>
<tr>
<td></td>
<td>EXACT: BMC has tested the SNMP queries used to discover this device on a</td>
</tr>
<tr>
<td></td>
<td>device of exactly this model.</td>
</tr>
<tr>
<td></td>
<td>SIMILAR: BMC has tested the SNMP queries used to discover this device on</td>
</tr>
<tr>
<td></td>
<td>a device that is similar, but not identical, to this device. For</td>
</tr>
<tr>
<td></td>
<td>example, the queries might have been tested on a device from the same</td>
</tr>
<tr>
<td></td>
<td>family of devices from the same vendor, but not on this exact model.</td>
</tr>
<tr>
<td>Serial ID</td>
<td>The serial number of this management controller. For example, 417P95J.</td>
</tr>
<tr>
<td>Name</td>
<td>The sysname of this management controller. For example, IDRAC-C1KN95J.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>The SNMP System Object ID of the management controller. For example, 1.3.6.1.4.1.674.10892.2</td>
</tr>
<tr>
<td>OS Class</td>
<td>The OS class of this management controller. For example, UNIX.</td>
</tr>
<tr>
<td>OS Type</td>
<td>The OS type of this management controller. For example, GNU/Linux.</td>
</tr>
</tbody>
</table>
Field name | Details
--- | ---
OS Version | The OS version of this management controller. For example, 2.6 kernel.

Network Interfaces | Provides the following details for each network interface:
- **Name**: The name of the network interface.
- **Description**: A description of the network interface.
- **MAC Address**: The MAC address of the network interface which is also a link to the page for that network interface.
- **Details**: The interface speed. A red x icon is displayed where an interface has some mismatch (speed, duplex, or negotiation) with the network device that it is connected to.
- **Connected To**: The network device that the network interface is connected to. A red x icon is displayed where the network device port has some mismatch (speed, duplex, or negotiation) with the interface connected to it.
- **Show details**: Click *Show Details* to reveal more information on the network interfaces. Where a mismatch has been flagged in the **Connected To** or **Details** fields, the reason for this is highlighted.
- **Show All nnn**: Click *Show All nnn* to show all of the available networking interfaces.

IPv4 Addresses | The name and IPv4 address of the network device. Click *Show Details* to see Netmask and Broadcast information.

IPv6 Addresses | The name and IPv6 address of the network device. Click *Show Details* to see Prefix information.

Managed Host | Details of the host that the management controller is managing.
- **Name**: The name of the host.
- **OS**: The OS of the host.
- **Hardware Vendor**: The hardware vendor for the host.
- **Virtual**: Whether or not the host is virtual.
- **Partition**: The name of the host.
- **Scanned via**: The IP address through which the host was scanned.

### Viewing a mainframe

A mainframe object in BMC Atrium Discovery represents a mainframe computer, identified by an IP address range. Mainframe objects are set up automatically by the BMC Atrium Discovery Discovery and Reasoning process.

- To view a mainframe
- Default Section
- MFParts
- Identity
- Hardware
- Inference

#### To view a mainframe

1. Click the Mainframes link in the Infrastructure Summary section of the Infrastructure page.
2. Select a mainframe from the list.
The BMC Atrium Discovery Mainframe page groups the field names in a logical format. The page is divided into separate groups of information which you can collapse or expand to view as required.

The information fields for a mainframe object are arranged in the following groups:

- **Default Section** — General information on the mainframe such as its name and vendor. See Default Section below.
- **MFParts** — Information on LPARs contained in the mainframe. See MFParts for details.
- **Identity** — Serial number and description of the mainframe. See Identity for details.
- **Hardware** — Information on the mainframe model and number of processors. See Hardware for details.
- **Inference** — Inference details. See Inference for details.

**Default Section**

A list of information fields which can be displayed in the Default Section group for a mainframe object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hostname</td>
<td>The hostname of the mainframe computer.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The vendor of the mainframe computer.</td>
</tr>
</tbody>
</table>

**MFParts**

MFParts represent the LPARs of the mainframe computer. The links to the LPARs are listed in this section. See MF Part page for more information.

**Identity**

A list of information fields that can be displayed in the Identity group for a mainframe object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial</td>
<td>The serial number of the mainframe computer.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the mainframe computer.</td>
</tr>
</tbody>
</table>
Hardware

A list of information fields that can be displayed in the Hardware group for a mainframe object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>The model name of the mainframe computer.</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>The number of processors in the mainframe computer.</td>
</tr>
</tbody>
</table>

Inference

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Lists Discovery Accesses for this mainframe, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run of which this is a part, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess Page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.</td>
</tr>
</tbody>
</table>

MF Part page

The BMC Atrium Discovery MF Part page provides details of the MF Part. MF Parts are LPARs and might be native or virtual. The MF Parts page groups the field names in a logical format. The page is divided into separate groups of information which you can collapse or expand to view as required.

- Default Section
- Identity
- Operating System
- Storage
- Inference
- CMDB Synchronization

Default Section

A list of information fields that can be displayed in the Default Section group for a mainframe object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the LPAR.</td>
</tr>
<tr>
<td>Model</td>
<td>The model of the LPAR.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The LPAR vendor.</td>
</tr>
<tr>
<td>Virtual</td>
<td>A flag indicating whether the MFPart is virtual.</td>
</tr>
<tr>
<td>Partition</td>
<td>A flag indicating whether the MFPart is a partition.</td>
</tr>
<tr>
<td>Software Instances</td>
<td>Software Instances running on this MFPart.</td>
</tr>
</tbody>
</table>
Field name | Details
---|---
Mainframe | A link to the object page for the containing mainframe computer.
Sysplex | A link to the object page for the sysplex. This is modeled as a Cluster node.
Coupling Facility | A link to the coupling facility that the LPAR is using.

Identity

The full list of information fields which can be displayed in the Identity group for a mainframe object are shown in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Type</td>
<td>The serial number of the mainframe computer.</td>
</tr>
<tr>
<td>VM ID</td>
<td>The description of the mainframe computer.</td>
</tr>
<tr>
<td>Virtual</td>
<td>States whether the LPAR is virtual.</td>
</tr>
</tbody>
</table>

Operating System

The full list of information fields which can be displayed in the Operating System group for a mainframe object are shown in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered OS</td>
<td>The discovered OS, for example, z/OS SP7.1.1..</td>
</tr>
<tr>
<td>Discovered OS Class</td>
<td>The discovered OS class (for example, Mainframe).</td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>The discovered OS type (for example, z/OS).</td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>The discovered OS type (for example, SP7.1.1).</td>
</tr>
</tbody>
</table>

Storage

The full list of information fields which can be displayed in the Storage group for a mainframe object are shown in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Collection</td>
<td>Link to the storage collections used by this MF Part.</td>
</tr>
</tbody>
</table>

Inference

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Lists Discovery Accesses for this mainframe, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run of which this is a part, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess Page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.</td>
</tr>
</tbody>
</table>
CMDB Synchronization

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last successful CMDB sync</td>
<td>The time and date at which this MFPart was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time and date at which this MFPart was last unsuccessfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>The time (in seconds) spent performing the last CMDB synchronization for this MFPart.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>The number of CIs corresponding to this MFPart at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>The number of relationships to CIs corresponding to this MFPart at the last CMDB synchronization.</td>
</tr>
</tbody>
</table>

Coupling facility page

The BMC Atrium Discovery coupling facility page provides details of the mainframe computer's coupling facility, the component of the mainframe that enables multiple LPARs to access the same data.

A list of information fields that can be displayed in the coupling facility page are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The coupling facility name.</td>
</tr>
<tr>
<td>Internal Storage</td>
<td>The amount of internal storage (in bytes) on the coupling facility.</td>
</tr>
<tr>
<td>Resource Manager Policy</td>
<td>Resource Manager Policy for this coupling facility.</td>
</tr>
<tr>
<td>Member MFParts</td>
<td>A link any MFParts (currently LPARs) that are dependent on this coupling facility.</td>
</tr>
</tbody>
</table>

Sysplex page

The BMC Atrium Discovery Sysplex page provides details of the mainframe computers sysplex. The sysplex is the component of the mainframe which manages the contained LPARs.

A list of information fields that can be displayed in the Sysplex page are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The sysplex name.</td>
</tr>
<tr>
<td>Type</td>
<td>The sysplex is modeled using a cluster node. This field refers to the type of cluster which for a sysplex is always Sysplex.</td>
</tr>
<tr>
<td>Internal Identifier</td>
<td>The internal identifier of the sysplex.</td>
</tr>
<tr>
<td>Description</td>
<td>A description of the sysplex. Currently this is syplexname on mainframename.</td>
</tr>
<tr>
<td>Member MFParts</td>
<td>A link to any MFParts (currently LPARs) that are part of this sysplex.</td>
</tr>
<tr>
<td>Member Coupling Facilities</td>
<td>A link to any coupling facilities that are part of this sysplex.</td>
</tr>
</tbody>
</table>
Viewing a network device

A network device object in BMC Atrium Discovery represents a switch, a router, a firewall, a SAN switch and so forth. Network devices are discovered using SNMP credentials. A valid credential is required to create a Network Device node.

- To view a network device
- General details
- Identity
- Infrastructure
- Inference
- CMDB Synchronization
- To view an imported network device
- To run network device reports
- Network device and host mismatch

⚠️ Note

You should either discover or import network devices. Doing both will result in duplicates where a discovered network device is also imported.

To view a network device

1. Click the **Infrastructure** tab, and then click the **Network Devices** icon in the Infrastructure Summary section.
   - The Network Devices page is displayed.
2. Select a network device from the list.

A typical example of a discovered network device is illustrated in the following screen.

![Network Device Example](image)

The information fields for a discovered network device object are arranged in the following groups:
• General Details: General information on the network device such as its name, type, and model. See General Details below.
• Identity: Information on domain names and IP addresses. See Identity for details.
• Operating System: The Operating system information for this network device. See Operating System for details.
• Infrastructure: Information on the services that the network device provides. See Infrastructure for details.
• Inference: The inference details. See Inference for details.

⚠️ Note

You can also display provenance information by clicking the Show Provenance button. Provenance information is meta-information describing how the other information came to exist. For more information, see Provenance Information.

General details

The full list of information fields which can be displayed in the General Details group for a network device are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the network device. If no name is discovered then this is the IP address of the device.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the network device, such as Switch, Router, and so forth.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The vendor of the network device.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name of the network device.</td>
</tr>
<tr>
<td>Status</td>
<td>Whether the exact SNMP device has been tested by BMC. This can be either: EXACT: BMC has tested the SNMP queries used to discover this device on a device of exactly this model. SIMILAR: BMC has tested the SNMP queries used to discover this device on a device that is similar, but not identical, to this device. For example, the queries might have been tested on a device from the same family of devices from the same vendor, but not on this exact model.</td>
</tr>
</tbody>
</table>

Identity

The full list of information fields that can be displayed in the Identity group for a network device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial ID</td>
<td>The serial number of the network device.</td>
</tr>
<tr>
<td>System Name</td>
<td>The sysname of the network device.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>The SNMP System Object ID of the network device.</td>
</tr>
</tbody>
</table>
Operating System

The full list of information fields that can be displayed in the Operating System group for a network device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered OS Class</td>
<td>The discovered OS class for the network device (for example, Embedded).</td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>The discovered OS type for the network device (for example, IOS).</td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>The discovered OS version for the network device (for example, 12.1).</td>
</tr>
<tr>
<td>Discovered OS Vendor</td>
<td>The vendor of the discovered OS (for example, Cisco).</td>
</tr>
</tbody>
</table>

Infrastructure

The full list of information fields that can be displayed in the Infrastructure group for a network device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Interfaces</td>
<td>Provides the following details for each network interface:</td>
</tr>
<tr>
<td></td>
<td>• IP Address: The IP address, also a link to the page for that IP address.</td>
</tr>
<tr>
<td></td>
<td>• MAC Address: The MAC address of the network interface which is also a link to the page for that network interface.</td>
</tr>
<tr>
<td></td>
<td>• Details: The interface number (eth0, eth1) which is also a link to the page for that network interface. A red x icon is displayed where an interface has some mismatch (speed, duplex, or negotiation) with the network device that it is connected to.</td>
</tr>
<tr>
<td></td>
<td>• Connected To: The network device that the network interface is connected to. A red x icon is displayed where the network device port has some mismatch (speed, duplex, or negotiation) with the interface connected to it.</td>
</tr>
<tr>
<td></td>
<td>• Show details: Click Show Details to reveal more information on the network interfaces. Where a mismatch has been flagged in the Connected To or Details fields, the reason for this is highlighted.</td>
</tr>
<tr>
<td></td>
<td>• Show All: Click Show All nnn to show all of the available networking interfaces.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>The name and IPv4 address of the network device. Click Show Details to see Netmask and Broadcast information.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>The name and IPv6 address of the network device. Click Show Details to see Prefix information.</td>
</tr>
<tr>
<td>Fibre Channel Ports</td>
<td>Provides the following details for the device's fibre channel ports, for example, for a SAN Switch:</td>
</tr>
<tr>
<td></td>
<td>• Port Name: The name of the fibre channel port. For example, FC Port 0/0.</td>
</tr>
<tr>
<td></td>
<td>• WWPN: The WWPN.</td>
</tr>
<tr>
<td></td>
<td>• Type: The type of fibre channel port. For example, Fabric or Generic Fabric.</td>
</tr>
<tr>
<td></td>
<td>• State: The state of the port. For example, operational.</td>
</tr>
<tr>
<td></td>
<td>• Speed: The port speed.</td>
</tr>
</tbody>
</table>

Inference

The full list of information fields which can be displayed in the Inference group for a network device object are displayed in the following table.
Field name | Details
---|---
Discovery Access | Lists Discovery Accesses for the network device, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run that the scan is part of, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.

CMDB Synchronization

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last successful CMDB sync</td>
<td>The time and date when the network device was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time and date when the network device was last unsuccessfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>The time (in seconds) spent performing the last CMDB synchronization for the network device.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>The number of CIs corresponding to the network device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>The number of relationships to CIs corresponding to the network device at the last CMDB synchronization.</td>
</tr>
</tbody>
</table>

To view an imported network device

A network device object in BMC Atrium Discovery can also represent an imported switch. A typical example of an imported network device follows.

Information fields for a typical imported network device

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the network device.</td>
</tr>
<tr>
<td>Port Details</td>
<td>The following details are provided for each port on the network device: Port, Speed, Duplex, Negotiation, IP Address, Connected Host, and Description. This also highlights any performance affecting mismatches. See #Network device/host mismatch for more information.</td>
</tr>
<tr>
<td>Connected Hosts</td>
<td>List of connected hosts. The following details are provided for each connected host. Name, Discovered OS, Hardware Vendor, Virtual (whether the host is a virtual host), Organisational Unit, Location, and Status.</td>
</tr>
<tr>
<td>Data Quality Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

To run network device reports

A number of context-sensitive reports are available for network devices. See Switch Reports for additional information and examples of these network device-related reports.
Network device and host mismatch

The View Object page of a network device displays attributes and relationships that are already set for that network device. In the Port Details row, it also displays information on the connection settings for the ports on the network device.

The Port Details row is divided into fields. Of these, the speed, duplex, and negotiation fields reflect whether there is a performance affecting mismatch, or whether insufficient information is available to establish whether a mismatch exists.

If a mismatch is detected, or information is missing so that the correct configuration cannot be confirmed, then the field is filled with a background color.

- Where a mismatch is detected, the field is filled with a red background.
- Where there is insufficient information to confirm the correct configuration, the field is filled with a yellow background.

You can view details of the mismatch or the missing data details by clicking the + symbol in a colored field. The details are displayed in a popup window. This is shown below:

This screen illustrates the popup window that displays when you view the details of missing data.

⚠️ Note

Hyper-V Windows virtual machines (VMs) always report NIC speed as 10 GBps regardless of the actual speed. Consequently, for ports connected to Hyper-V Windows VMs, the network device and switch mismatch field shows incorrect results.

Managing network device virtual interface discovery

BMC Atrium Discovery has the ability to discover network device virtual interfaces. By default, this ability is disabled. You can manage network device virtual interface discovery by using the `tw_options` command.
Use of the `tw_options` command is strictly restricted to managing network device virtual interface discovery and managing attachments only. Unless explicitly instructed by BMC Support, you must not change any other `tw_options` options as it may result in unexpected BMC Atrium Discovery behavior.

**Enabling network device virtual interface discovery**

To enable network device virtual interface discovery:

1. Log in to the appliance command line as the `tideway` user.
2. You must use the system user, or another user belonging to the system or admin group to be able to enable network device virtual interface discovery.
   - If using the system user, enter:
     ```bash
     [tideway@localhost ~]$ tw_options NETWORK_DISCOVERY_VIRTUAL_INTERFACES=True
     ```
   - If you are using another user (other than the system user) belonging to the system or admin group, enter:
     ```bash
     tw_options -u <user name> NETWORK_DISCOVERY_VIRTUAL_INTERFACES=True
     ```
3. When prompted, depending on the command you have run in the previous step, enter the system or user's UI password.
4. Exit from the command line prompt.

**Disabling network device virtual interface discovery**

If network device virtual interface discovery has been enabled and you want to disable it, you must:

1. Log in to the appliance command line as the `tideway` user.
2. You must use the system user, or another user belonging to the system or admin group to be able to disable network device virtual interface discovery.
   - If using the system user, enter:
     ```bash
     [tideway@localhost ~]$ tw_options NETWORK_DISCOVERY_VIRTUAL_INTERFACES=False
     ```
   - If you are using another user (other than the system user) belonging to the system or admin group, enter:
3. When prompted, depending on the command you have run in the previous step, enter the system or user's UI password.
4. Exit from the command line prompt.

### Viewing load balancers

Load balancer information is **discovered** by BMC Atrium Discovery using SNMP. BMC Atrium Discovery identifies a hardware or virtualized Application Delivery Controller (ADC) that has load balancer options enabled as load balancer. When such a device is found, BMC Atrium Discovery creates a network device node and triggers the load balancer discovery patterns to discover the associated load balancing components. These components include pools, hosts and services, which are represented in the BMC Atrium Discovery UI and are described in the following pages:

- Viewing a load balancer device
- Viewing a load balancer group
- Viewing a load balancer instance
- Viewing a load balancer pool
- Viewing a load balancer member
- Viewing a load balancer service

### Viewing a load balancer device

A load balancer device in BMC Atrium Discovery represents a hardware or virtualized Application Delivery Controller (ADC) that has load balancer options enabled.

To view a load balancer device

1. Click the **Infrastructure** tab.
2. Click **All Load Balancer Devices** in the Load Balancer Reports section.
3. Select a load balancer device from the list.

The information about a load balancer device is exactly the same as the information about a network device. For more information about load balancer device details, see [Viewing a network device](#).

### Viewing a load balancer group

A load balancer group in BMC Atrium Discovery represents a group of connected load balancers.

You can view a load balancer group from a Load Balancer Instance page.

To view the load balancer group

Open the load balancer instance page:

1. Click the **Infrastructure** tab.
2. Click **Load Balancer Instances** in the Infrastructure Summary section.
3. Select a load balancer instance from the list.
4. Click the **Failover group** link.

A typical example of a discovered load balancer group is illustrated in the following screen:

![Load Balancer Group: F5 BIG-IP Group of lon-vm-f5bigip.tideway.com, lon-vm-f5bigip-2.tideway.com and 1 more](image)

The information fields for a discovered load balancer group are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the load balancer group. For example, F5 BIG-IP Group of lon-vm-f5bigip.tideway.com, lon-vm-f5bigip-2.tideway.com and 1 more.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the load balancer group. For example, F5 BIG-IP Group.</td>
</tr>
<tr>
<td>Failover type</td>
<td>Type of failover the load balancer group uses. For example, Active-Standby, Active-Active, Service Clustering.</td>
</tr>
<tr>
<td>Load Balancers</td>
<td>The list of links to load balancer instances that are a part of the group. To see the load balancer instance page, click the corresponding link. For more information, see <a href="#">Viewing load balancer instance</a>.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>Information about data completeness issues.</td>
</tr>
</tbody>
</table>

**To view the Load Balancers in failover pairs or clusters report**

To find the load balancer instances that are configured to be a part of the failover group:

1. Click the **Infrastructure** tab.
2. Click **Load Balancers in failover pairs or clusters** in the Load Balancer Reports section.
3. Select a load balancer group from the list.

Following is a typical example of the Load Balancers in failover pairs or clusters page:
Viewing a load balancer instance

A load balancer instance in BMC Atrium Discovery represents a load balancer instance configured on a network device.

To view a load balancer instance:

1. Click the **Infrastructure** tab.
2. Click **Load Balancer Instances** in the Infrastructure Summary section.
3. Select a load balancer instance from the list.

   A typical example of a discovered load balancer instance is illustrated in the following screen.

The information fields for a discovered load balancer instance are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the load balancer instance. For example, F5 BIG-IP on lon-vm-f5bigip.tideway.com.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of load balancer instance. For example, F5 BIG-IP.</td>
</tr>
<tr>
<td>Version</td>
<td>The load balancer instance version. For example, 11.5.1-0.0.110-Final.</td>
</tr>
<tr>
<td>Device</td>
<td></td>
</tr>
<tr>
<td>Field name</td>
<td>Details</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>The link to the device where a load balancer instance is running, for example, lon-vm-f5bigip.tideway.com. To see the load balancer device page, click the corresponding table row. For more information, see Viewing load balancer device.</td>
</tr>
<tr>
<td>Failover type</td>
<td>Type of failover the load balancer instance uses. For example, Active-Standby, Active-Active, Service Clustering.</td>
</tr>
<tr>
<td>Failover state</td>
<td>State of the failover for this load balancer instance. For example, Active or Standby.</td>
</tr>
<tr>
<td>Failover group</td>
<td>The link to a load balancer group that contains this instance. To see the load balancer group page, click the corresponding table row. For more information, see Viewing load balancer group.</td>
</tr>
<tr>
<td>Configuration</td>
<td>Displays the following configuration information:</td>
</tr>
<tr>
<td></td>
<td>• Failover state of the load balancer instance.</td>
</tr>
<tr>
<td></td>
<td>• List of related failover devices and their failover state. To see the failover device page, click the corresponding link. For more information, see Viewing load balancer device.</td>
</tr>
<tr>
<td></td>
<td>• Load balancer services available on the load balancer instance. To see the load balancer service page, click the corresponding link. For more information, see Viewing load balancer service.</td>
</tr>
<tr>
<td></td>
<td>• Load balancer pools that groups load balancer members on the load balancer instance. To see the load balancer pool page, click the corresponding link. For more information, see Viewing load balancer pool.</td>
</tr>
<tr>
<td></td>
<td>• Load balancer members (a physical server that acts as a provider of a service available to a load balancer instance). To see the load balancer member page, click the corresponding link. For more information, see Viewing load balancer member.</td>
</tr>
<tr>
<td></td>
<td>• Hosts where a load balancer services of the instance are running. To see the host page, click the corresponding link.</td>
</tr>
</tbody>
</table>

**Viewing a load balancer pool**

A load balancer pool in BMC Atrium Discovery represents a pool of members on a load balancer.

The total number of pools associated with a load balancer is displayed on the load balancer instance page.

To view a load balancer instance page:

1. Click the **Infrastructure** tab.
2. Click **Load Balancer Instances** in the Infrastructure Summary section.
3. Select the load balancer instance to view.
4. On a load balancer instance page, click on the link corresponding to Load Balancer Pools to view the Load Balancer Pool List page.

Following is a typical example of a Load Balancer Pool List page:
The load balancer pools are displayed in rows and the following fields are displayed for each pool:

- **Name**: The name of the pool.
- **Protocol**: The protocol pool uses.

Click on a row to view the corresponding Load Balancer Pool page. The following is a typical example of the Load Balancer Pool page:

The Load Balancer Pool page may display the following:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the load balancer pool. For example, /Common/Lync_2010.app/Lync_2010_front_end_ip_80_pool.</td>
</tr>
</tbody>
</table>
### Field name | Details
--- | ---
Load Balancer | The link to the load balancer instance that has this load balancer pool configured on it. To see the load balancer instance page, click the corresponding link. For more information, see Viewing load balancer instance.
Service | The link to the load balancer service that uses this load balancer pool. To see the load balancer service page, click the corresponding link. For more information, see Viewing load balancer service.
Members | A table that contains the following information about the members of this load balancer pool:
- **IP Address**
- **Port**
- **Host**

To see the load balancer member page, click the corresponding row. For more information, see Viewing load balancer member.

### Viewing a load balancer member

The load balancer member in BMC Atrium Discovery represents a physical server that acts as a provider of a service available to a load balancer.

You can find a load balancer member either in Load Balancers Members reports, or in a Load Balancer Instance page.

- Click the **Infrastructure** tab, and then:
  - For the **Load Balancer Members not linked to a Host** report, click **Load Balancer Members not linked to a Host** in the Load Balancer Reports section.
  - For the **Load Balancer Members not linked to software** report, click **Load Balancer Members not linked to software** in the Load Balancer Reports section.
  - For the load balancer instance:
    - Click **Load Balancer Instances** in the Infrastructure Summary section.
    - Select a load balancer instance from the list.
    - In Configuration section, click on the link in a Members column.

Following is a typical example of the Load Balancer Member List page:
The load balancer members are displayed in rows and the following fields are displayed for each member:

- IP address
- Port
- Host

Click on a row to view the corresponding Load Balancer Member page. Following is a typical example of the Load Balancer Member page:

The Load Balancer Member page may display the following:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the Load Balancer Member. For example, /Common/137.72.93.36:135.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address of the service. For example, 168.192.1.1.</td>
</tr>
<tr>
<td>Port</td>
<td>The port that the service is using. For example, 135.</td>
</tr>
<tr>
<td>Pools</td>
<td>The list of links to the Load Balancer Pools that the Load Balancer Member is in. To see the load balancer pool page, click the corresponding link. For more information, see Viewing load balancer pool.</td>
</tr>
<tr>
<td>Host</td>
<td>The link to the host that implements the service.</td>
</tr>
</tbody>
</table>
Viewing a load balancer service

A load balancer service in BMC Atrium Discovery represents a virtual server on a load balancer. The total number of services associated with a load balancer is displayed on the load balancer instance page.

To view a load balancer services:

1. Click the Infrastructure tab.
2. Click Load Balancer Instances in the Infrastructure Summary section.
3. Select the load balancer instance to view.
4. On a Load Balancer Instance page, click on the link corresponding to Load Balancer Services to view the Load Balancer Service List page. Following is a typical example of the Load Balancer Service List page:

![Load Balancer Service List](image)

The load balancer services are displayed in rows and the following fields are displayed for each service:

- Name
- IP address
- Port

Click on a row to view the corresponding Load Balancer Service page. Following is a typical example of the Load Balancer Service page.
The Load Balancer Service page may display the following:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the load balancer service. For example, /Common/jp_virtual_server_2.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address of the service.</td>
</tr>
<tr>
<td>Port</td>
<td>The port service is using.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of load balancer service. For example, poolbased.</td>
</tr>
<tr>
<td>State</td>
<td>The status of load balancer service. For example, enabled.</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>The link to the load balancer instance the service is running on. To see the load balancer instance page, click the corresponding link. For more information, see Viewing load balancer instance.</td>
</tr>
</tbody>
</table>

Viewing a network printer

A network printer object in BMC Atrium Discovery represents any printer in an estate. Network printers are discovered using SNMP credentials.

Printer discovery definitions can be added or modified in the same manner as Network Devices through a Technology Knowledge Update (TKU).

- To view a network printer
  - General Details
  - Identity
  - Operating System
  - Infrastructure
  - Inference

To view a network printer

To view a printer, click the Printer icon in the Infrastructure Summary section of the Infrastructure tab, and select a printer from the list.
A typical example of a discovered network printer is illustrated in the following screen.

The information fields for a discovered network printer object are arranged in the following groups:

- **General Details**: General information about the network printer such as its name, vendor, and model. See General Details below.
- **Identity**: Information about domain names and IP addresses. See Identity for details.
- **Operating System**: The operating system information for the network printer. See Operating System for details.
- **Infrastructure**: Information about the services that the network printer provides. See Services for details.
- **Inference**: The inference details. See Inference for more information.

⚠️ **Note**

You can also display provenance information by clicking Show Provenance. Provenance information is meta-information describing how the other information became available. For more information, see Provenance Information.

### General Details

The full list of information fields which can be displayed in the General Details group for a network printer are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the network printer.</td>
</tr>
<tr>
<td>Type</td>
<td>The device type. For a printer, this is Printer.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The vendor of the network printer.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name of the network printer.</td>
</tr>
<tr>
<td>Testing Status</td>
<td>Whether the exact printer has been tested by BMC. This can be either:</td>
</tr>
<tr>
<td></td>
<td>EXACT: BMC has tested the SNMP queries used to discover this device on a device of exactly this model.</td>
</tr>
<tr>
<td></td>
<td>SIMILAR: BMC has tested the SNMP queries used to discover this device on a device that is similar, but not identical, to this device. For example, a duplex version of a single side printer which is validated.</td>
</tr>
</tbody>
</table>
Identity
The full list of information fields that can be displayed in the Identity group for a network printer object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial ID</td>
<td>The serial number of the network printer.</td>
</tr>
<tr>
<td>System Name</td>
<td>The <code>sysname</code> of the network printer.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>The SNMP System Object ID of the network printer.</td>
</tr>
</tbody>
</table>

Operating System
The full list of information fields that can be displayed in the Operating System group for a network device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered OS Class</td>
<td>The discovered OS class for the network printer (for example, Embedded).</td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>The discovered OS type for the network printer (for example, JetDirect).</td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>The discovered OS version for the network printer (for example, 22.01).</td>
</tr>
<tr>
<td>Discovered OS Vendor</td>
<td>The vendor of the discovered OS (for example, HP).</td>
</tr>
</tbody>
</table>

Infrastructure
The full list of information fields that can be displayed in the Infrastructure group for a network printer object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Network Interfaces| Provides the following details for each network interface:  
• IP Address: The IP address, also a link to the page for that IP address.  
• MAC Address: The MAC address of the network interface, also a link to the page for that network interface.  
• Details: The interface number (eth0, eth1) which is also a link to the page for that network interface. A red x icon is displayed where an interface has some mismatch (speed, duplex, or negotiation) with the network device that it is connected to.  
• Connected To: The network device that the network interface is connected to. A red x icon is displayed where the network device port has some mismatch (speed, duplex, or negotiation) with the interface connected to it.  
• Show details: Click Show Details to reveal more information on the network interfaces. Where a mismatch has been flagged in the Connected To or Details fields, the reason for this is highlighted.  
• Show All `nnn`. Click Show All `nnn` to show all of the available network interfaces. |
| IPv4 Addresses    | The name and IPv4 address of the network printer. Click Show Details to see Netmask and Broadcast information.                                         |
| IPv6 Addresses    | The name and IPv6 address of the network printer. Click Show Details to see Prefix information.                                                     |
Inference

The full list of information fields that can be displayed in the Inference group for a network printer object are displayed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Lists Discovery Accesses for the network printer, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run that the scan is part of, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess Page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.</td>
</tr>
</tbody>
</table>

Viewing an SNMP managed device

An SNMP managed device object in BMC Atrium Discovery can represent a rack system, UPS, network analysis module, environmental monitoring unit, or other device which supports SNMP management. SNMP managed devices are discovered using SNMP credentials. A valid credential is required to create an SNMP Managed Device node.

- To view an SNMP managed device
- General Details
- Identity
- Ownership & Location
- Operating System
- Infrastructure
- Inference
- CMDB Synchronization

To view an SNMP managed device

1. Click the Infrastructure tab, and then click SNMP Managed Devices in the Infrastructure Summary section.
   The SNMP Managed Devices page is displayed.
2. Select an SNMP managed device from the list.

A typical example of a discovered SNMP managed device is illustrated in the following screen.
This screen illustrates an example of a discovered SNMP managed device.

The information fields for a discovered SNMP managed device object are arranged in the following groups:

- **General Details**: General information on the device such as its name, type, and model. See General Details below.
- **Identity**: Information on Serial number, system name, system object ID. See Identity for details.
- **Ownership & Location**: Information on the ownership and location of the host. See Ownership & Location for details.
- **Operating System**: The operating system information for this SNMP managed device. See Operating System for details.
- **Infrastructure**: Information on the network interfaces, and IP addresses (IPv4 and IPv6) that the SNMP managed device uses. See Infrastructure for details.
- **Inference**: The inference details. See Inference for details.
- **CMDB Synchronization**: Information on the last CMDB synchronization operations and the number of CIs corresponding to this SNMP managed device at the last CMDB synchronization.

⚠️ **Note**

You can also display provenance information by clicking the Show Provenance button. Provenance information is meta-information describing how the other information came to exist. For more information, see Provenance Information.

**General Details**

The full list of information fields which can be displayed in the General Details group for a SNMP managed device are described in the following table.
### Field name

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the SNMP managed device. If no name is discovered then this is the IP address of the device.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of SNMP managed device, such as UPS, Network Infrastructure, or Remote Access Controller.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The vendor of the SNMP managed device.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name of the SNMP managed device.</td>
</tr>
<tr>
<td>Status</td>
<td>Whether the exact SNMP device has been tested by BMC. This can be either: EXACT: BMC has tested the SNMP queries used to discover this device on a device of exactly this model. SIMILAR: BMC has tested the SNMP queries used to discover this device on a device that is similar, but not identical, to this device. For example, the queries might have been tested on a device from the same family of devices from the same vendor, but not on this exact model. Customer: The queries used to discover this device were contributed by a member of the BMC Atrium Discovery community.</td>
</tr>
</tbody>
</table>

### Identity

The full list of information fields that can be displayed in the Identity group for a SNMP managed device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial ID</td>
<td>The serial number of the SNMP managed device.</td>
</tr>
<tr>
<td>System Name</td>
<td>The sysname of the SNMP managed device.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>The SNMP System Object ID of the SNMP managed device.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>If the endpoint used to discover this SNMP managed device was previously identified as a different entity, a link to that entity and a link to a comparison with that entity are provided.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>If the endpoint used to discover this SNMP managed device is now identified as a different entity, a link to that entity and a link to a comparison with that entity are provided.</td>
</tr>
</tbody>
</table>

### Ownership & Location

A list of information fields that can be displayed in the Ownership & Location group for a host object are described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizational Unit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>The person or owner responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>The person or owner responsible for this element from an IT perspective.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>The person or owner responsible for the support of this element.</td>
</tr>
<tr>
<td>Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Status</td>
<td>The lifecycle status of this element.</td>
</tr>
</tbody>
</table>
Operating System

The full list of information fields that can be displayed in the Operating System group for a SNMP managed device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovered OS Class</td>
<td>The discovered OS class for the SNMP managed device (for example, Embedded).</td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>The discovered OS type for the SNMP managed device (for example, APC).</td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>The discovered OS version for the SNMP managed device (for example, 3.8.6).</td>
</tr>
<tr>
<td>Discovered OS Vendor</td>
<td>The vendor of the discovered OS (for example, APC).</td>
</tr>
</tbody>
</table>

Infrastructure

The full list of information fields that can be displayed in the Infrastructure group for a SNMP managed device object are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Interfaces</td>
<td>Provides the following details for each network interface:</td>
</tr>
<tr>
<td></td>
<td>• Name: The name of the network interface, also a link to the page for that network interface.</td>
</tr>
<tr>
<td></td>
<td>• Description: Description of the network interface.</td>
</tr>
<tr>
<td></td>
<td>• MAC Address: The MAC address of the network interface which is also a link to the page for that network interface.</td>
</tr>
<tr>
<td></td>
<td>• Details: The interface number (eth0, eth1) which is also a link to the page for that network interface. A red x icon is displayed where an interface has some mismatch (speed, duplex, or negotiation) with the network device that it is connected to.</td>
</tr>
<tr>
<td></td>
<td>• Connected To: The network device that the network interface is connected to. A red x icon is displayed where the network device port has some mismatch (speed, duplex, or negotiation) with the interface connected to it.</td>
</tr>
<tr>
<td></td>
<td>• Show details: Click Show Details to reveal more information on the network interfaces. Where a mismatch has been flagged in the Connected To or Details fields, the reason for this is highlighted.</td>
</tr>
<tr>
<td></td>
<td>• Show All: Click Show All nnn to show all of the available networking interfaces.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>The name, IPv4 address, and subnet of the SNMP managed device. Click Show Details to see Netmask and Broadcast information.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>The name, IPv6 address, and subnet of the SNMP managed device. Click Show Details to see Prefix information.</td>
</tr>
</tbody>
</table>

Inference

The full list of information fields which can be displayed in the Inference group for a SNMP managed device object are displayed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Lists Discovery Accesses for the SNMP managed device, grouped by scan date. When an endpoint is scanned, a Discovery Access node is created. The Discovery Access node records information such as the start time, end time, the Discovery Run that the scan is part of, and the previous Discovery Access to simplify troubleshooting. Clicking a Discovery Access displays the DiscoveryAccess page for that Discovery Access. Where a Discovery Access is about to be deleted as part of DDD aging, a trash can icon is displayed next to the Discovery Access.</td>
</tr>
</tbody>
</table>
CMDB Synchronization

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last successful CMDB sync</td>
<td>The time and date when the SNMP managed device was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time and date when the SNMP managed device was last unsuccessfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>The time (in seconds) spent performing the last CMDB synchronization for the SNMP managed device.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>The number of CIs corresponding to the SNMP managed device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>The number of relationships to CIs corresponding to the SNMP managed device at the last CMDB synchronization.</td>
</tr>
</tbody>
</table>

Supported SNMP devices

The following classes of SNMP device are supported. The links below link to Configipedia pages which provide up to date lists of supported devices of each class. Click the SNMP Devices icon on the Administration page for an definitive list of the supported devices on your appliance.

- Network devices
- Printers
- SNMP managed devices

Monthly TKU releases might increase the SNMP device coverage.

⚠️ Note

If you need BMC Atrium Discovery to support a new SNMP device, you should first try to create a recognition rule for it. This is described in Recognizing SNMP devices. You can also use the Device Capture capability to download a zipped MIB that you can forward to BMC Customer Support as part of a new support issue. BMC Atrium Discovery engineering will aim to incorporate support for the new device as a new feature in future TKU releases. You are recommended to raise the case reporting the unsupported SNMP device, including the zipped MIB as early as possible to maximize the chances of its inclusion in the earliest release possible.

BMC only provides integration support for SNMP-enabled devices that are either network printers or core networking infrastructure such as switches, routers, firewalls and load balancers. Other device classes, such as UPS devices, should be discovered through the use of recognition rules.
To view the SNMP devices supported by BMC Atrium Discovery

1. From the Discovery section on the Administration tab, click SNMP Devices. On the SNMP Devices page, supported devices are displayed organized in the following categories:
   - Vendor
   - Capability
   - Kind

2. Select a category to view the list of SNMP devices in that category. The following information is shown for each SNMP device:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>The name of the SNMP device.</td>
</tr>
<tr>
<td>Kind</td>
<td>The kind of SNMP device. This might be Network Device for switches, routers, firewalls and so on. For printers, the kind is Printer.</td>
</tr>
<tr>
<td>Capabilities</td>
<td>The capabilities, or purpose of the SNMP device. For example, Router, Switch, Layer 3 Switch, UPS, or Blade System.</td>
</tr>
<tr>
<td>sysObjectID</td>
<td>The SNMP sysObjectID.</td>
</tr>
<tr>
<td>Status</td>
<td>Whether the exact SNMP device has been tested by BMC. This can be either: EXACT: BMC has tested the SNMP queries used to discover this device on a device of exactly this model. SIMILAR: BMC has tested the SNMP queries used to discover this device on a device that is similar, but not identical, to this device. For example, the queries might have been tested on a device from the same family of devices from the same vendor, but not on this exact model.</td>
</tr>
<tr>
<td>Notes</td>
<td>Any additional useful notes on the device.</td>
</tr>
</tbody>
</table>

3. Click a device to display a list of all of methods that BMC Atrium Discovery might use to obtain data from a device, and the Object Identifiers accessed.
   Where an OID contains a table, a plus (+) icon is displayed.
4. Click the plus icon to reveal the table. Click the minus (-) icon to collapse the table as illustrated in the following screen.
This screen illustrates the methods discovery might use to obtain data from a device.

Not all devices support all methods.

**Recognizing SNMP devices**

Unknown SNMP devices are devices that can be accessed via SNMP, but BMC Atrium Discovery has insufficient knowledge of to discover fully. These might be switches, printers and so on that are currently not defined in TKU.

The SNMP Recognition Rules page enables you to create rules for unknown SNMP devices, so that they are identified in future by BMC Atrium Discovery. From these rules, BMC Atrium Discovery can create SNMPManagedDevice, Printer, and Network Device nodes. The node type and capabilities are used to choose the correct class (BMC_ComputerSystem, BMC_Printer, and so on) to model it in the CMDB.

To recognize a device appropriate SNMP credentials must be available, enabled, and valid for the IP address.

- Recognition approaches
- Limitations of recognized devices
- To view unrecognized SNMP devices
- To create recognition rules for a device
- Importing or exporting recognition rules
- To scan with a recognition rule
- Recognition rules and consolidation
Recognition approaches

An approach is a way of recognizing the device. BMC Atrium Discovery uses real world knowledge when offering possible recognition approaches. For example, if the sysObjectID states that the manufacturer is Xerox, then the device is almost certainly a printer. Similarly, Dell and HP printers always have printer in the sysDescr. The same practice is used for switches and load balancers. Where the device would be modeled as a Network Device node, you can always choose to create an SNMP Managed Device node instead.

In the sysDescr, you might be able to see information about the device. From there the most appropriate recognition approach might be obvious. For example, where the sysDescr begins "Dell 1350cnw Color Printer" and the approaches offered are:

- SNMP Managed Device (dell/generic)
- Printer (dell/printer)
- Network Device (dell/generic)
- Network Device (dell/powerconnect3000)

The approach most likely to be successful is "Printer (dell/printer)", although it is possible to use any listed.

Limitations of recognized devices

Where SNMP devices are recognized using these recognition rules, they are not "fully discovered" as they would be with a TKU pattern. Rather, they provide a means of recognizing known device types.

The approach determined is the most likely to succeed, but is not always correct. If the data has not been collected correctly then you need to use the alternative generic approach provided (for example, (dell/generic) in the example above). If that also fails, the device cannot be discovered without adding support for it to a TKU. In this case, you should use the device capture to download the zipped MIB and send it to the BMC Software Technology Update (TKU) team to request that support be included in BMC Atrium Discovery for that SNMP device.

The recognition approach does not work for some devices, as they do not support the generic parts of the MIB that BMC Atrium Discovery queries. Where no device is created, the methods required to create one might have failed, as the queries cannot be run against the unsupported generic parts of the MIB.

To view unrecognized SNMP devices

From the secondary navigation bar on the Discovery tab, click SNMP Recognition. The SNMP Recognition Rules page is displayed.
Shows unrecognized SNMP devices on the SNMP Recognition Rules page.

The **SNMP Recognition Rules** page shows the following information for each unrecognized SNMP device.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>The number of this type of unrecognized device that have been detected.</td>
</tr>
<tr>
<td>sysObjectID</td>
<td>The SNMP System Object ID of the SNMP managed device.</td>
</tr>
<tr>
<td>Sample sysDescr</td>
<td>An excerpt from the SNMP system description.</td>
</tr>
</tbody>
</table>
| Status     | **Unknown Device**: An unrecognized SNMP device that has no recognition rule created.  
**Pending Discovery**: A device for which a TKU has a rule, but has not been scanned since the TKU was loaded.  
**Rule Defined**: A rule is defined for that sysObjectID  
**Overridden by TKU**: A rule is defined for the sysObjectID but the loaded TKU also has one defined. The TKU rule takes precedence.  
**Generic sysObjectID**: Where multiple devices share the same sysObjectIDs, for example embedded devices using the NET-SNMP or UCD SNMP agent, they cannot be fully identified using this approach. Some of these devices can be identified by reference to other SNMP System Object IDs, though this is not user configurable. |
| Kind       | Kind of device. For example, SNMP Managed Device, Printer or Network Device. |
| Vendor     | The vendor of the SNMP managed device. |
| Model      | The model name of the SNMP managed device. |
| Actions    | An dynamic list of available actions. The following are displayed depending on the Status described above:  
**Create** — Open the **SNMP Recognition Rule** dialog and create recognition rules. Displayed where status is Unknown Device.  
When you have created a rule, the following actions are available:  
**Edit** — Open the **SNMP Recognition Rule** dialog to edit existing recognition rules. This action is removed when a TKU rule overrides a recognition rule.  
**Delete** — Delete an existing recognition rule.  
**Scan** — Scan the device using the recognition rule to identify it. |

To create recognition rules for a device

1. For the unrecognized device for which you want to create a rule, select **Create** from the **Actions** column.  
The **SNMP Recognition Rule** dialog is displayed.
1. Shows the dialog into which you enter data to create recognition rules.

2. Enter information on the device into the fields on the **SNMP Recognition Rule** dialog.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sysObjectID</td>
<td>A read only field showing the sysObjectID. Also provides a link to a new Google search page for the sysObjectID.</td>
</tr>
<tr>
<td>sysDescr</td>
<td>A read only field showing the sysDescr.</td>
</tr>
<tr>
<td>Approach</td>
<td>A drop down list showing available recognition approaches. Select the one most appropriate for the device. For more information, see approach.</td>
</tr>
<tr>
<td>Vendor</td>
<td>Enter the vendor of the device. This information can often be obtained from the sysDescr field above. You cannot save the recognition rule without entering a Vendor name.</td>
</tr>
<tr>
<td>Model</td>
<td>Enter the model name or number of the device. This information can often be obtained from the sysDescr field above. You cannot save the recognition rule without entering a Model name.</td>
</tr>
<tr>
<td>Capabilities</td>
<td>Select the device capabilities from the drop down list. For example, for the &quot;Dell 1350cnw Color Printer&quot;, select Print. Where a device has multiple capabilities, click Add another and select from the new drop down list. For example, a self-contained NAS might also have a Media Library so you should select both of those capabilities.</td>
</tr>
<tr>
<td>Testing</td>
<td>Select an IP address from the drop down list. This list is populated with the IP addresses that the unrecognized device has been reported on. Click Start to start the test. As the test progresses, the methods run are shown along with results and a success or failure icon.</td>
</tr>
</tbody>
</table>

3. Click **Save** to save the recognition rule.

4. The SNMP Recognition Rules page is displayed and the status of the rule is now Rule Defined.

**Importing or exporting recognition rules**

You can import or export recognition rules so you can transfer them between BMC Atrium Discovery instances.

**To export recognition rules**

1. From the SNMP Recognition Rules page, select the recognition rules that you want to export.
2. Select **Export Recognition Rules** from the **Actions** menu. The rule is exported and the UI displays a banner showing the number of rules exported and a download link.

3. Click the download link and save the zipped recognition rules file.

### To import recognition rules

1. From the SNMP Recognition Rules page, select **Import Recognition Rules** from the **Actions** menu. The UI displays the **Import SNMP Recognition Rules** dialog.
2. Click **Browse** and locate the zipped recognition rules file on your file system.
3. Select the zipped recognition rules file and click import.
   The SNMP recognition rules file is imported.

#### Warning

If an SNMP Recognition Rule is exported and subsequently imported to the same BMC Atrium Discovery machine, any changes made to the rule in the UI between the export and import are overwritten without warning.

### To scan with a recognition rule

After you have created a recognition rule, you can scan the IP address.

1. Click **Scan** from the Actions column. The **Add a New Run** dialog is displayed populated with the IP address to scan.
2. Click OK to scan the device. The Discovery page is displayed showing the **Currently Processing Runs** tab. When the run is completed, it is moved to the **Recent Runs** tab.
3. Click **Recent Runs**.
4. Click the entry on the **Recent Runs** tab to see the DiscoveryRun page for that run.
5. Click the entry in the Summary section of the DiscoveryRun page to see the new SNMP Managed Device or Printer.

If you delete a recognition rule that you have used to scan the device, the device is no longer unrecognized and does not appear on the SNMP Recognition Rules page. To create a new rule for the device, you must rescan it. As there is no recognition rule in place (it was deleted) the device is unrecognized again and appears in the SNMP Recognition Rules page.

### Recognition rules and consolidation

When you have developed SNMP recognition rules on a scanning appliance in a consolidating environment, you must also copy the recognition rules to any other scanning appliance in the consolidating environment. To do this:
1. From the SNMP Recognition Rules page, select the rules that you want to copy to the other scanning appliance.
2. Select **Export Recognition Rules** from the **Actions** menu.
   An information banner displays the number of rules that have been exported and provides a **Download** link.
3. Click the **Download** link and save the file onto your file system.

For every other scanning appliance:

1. Log into the scanning appliance and navigate to the SNMP Recognition Rules page.
2. Select **Import Recognition Rules** from the **Actions** menu.
3. Enter the file name of the exported rules file into the **Filename** dialog box, or click **Browse...** to locate the file.
4. Click **Import**.
5. The screen is refreshed and the imported rules shown against corresponding unrecognized devices.

You do not need to copy the rules to consolidation appliances.

**Capturing SNMP devices**

If an SNMP device is not supported, you can capture the device using BMC Atrium Discovery to dump the MIB of an SNMP agent. The MIB is used to create a .Zip archive file that is saved to a specified location on the appliance file system. The .Zip file can be sent to the BMC Software Technology Update (TKU) team to request that support be included in BMC Atrium Discovery for that SNMP device. You can start a capture from either the **Device Capture** tab or from Discovery Info nodes on the Device Info List page.

A valid SNMP credential must be available to capture an SNMP device.

- To capture unsupported devices from the Device Capture tab
- To capture unsupported devices from Discovery Info nodes

**To capture unsupported devices from the Device Capture tab**

1. From the Discovery section on the **Administration** tab, click **Device Capture**.
   The list of captured devices will display on the Device Captures page. If no devices were found, a banner displays stating that you must create and subsequently capture one. This is illustrated in the following screen.
2. If no devices are listed, select a node from the Discovery Access page and choose **Capture Device Data** from the **Actions** menu, as illustrated in the following screen.
Captured devices display on the Device Captures page and include the following detail:

- Endpoint
- Last Captured
- State
- Expected Manufacturer
- Expected Model
- Actions

3. From the Actions column, click the corresponding link to perform the following actions:
   - View Log: View the information about the capture (state, duration, and so forth) in a log file.
   - Edit: Revise the entered information for the capture (name, description, and so forth) in a Description field.
   - Capture: Run the capture (potentially overwriting any existing data). This is only available for captures that do not have a State of In Progress.
   - Download: Download the capture as a .zip file. This option is only available if the capture completed successfully.
   - Cancel: Stop an "In Progress" capture. A Capture Cancelled banner displays and an error will display in the log.
   - Delete: Remove the capture files.

The following screen illustrates the log page of a successful capture.

To capture unsupported devices from Discovery Info nodes

1. Select a node or nodes from the Device Info List page and choose **Capture Device Data** from the **Actions** menu.

Captured devices display on the Device Captures page and include the following detail:

- Endpoint
- Last Captured
- State
- Expected Manufacturer
- Expected Model
- Actions

2. From the Actions column, click the corresponding link to perform the following the actions:
- View Log: View the information about the capture (state, duration, and so forth) in a log file.
- Edit: Revise the entered information for the capture (name, description, and so forth) in a Description field.
- Capture: Run the capture (potentially overwriting any existing data). This is only available for captures that do not have a State of In Progress.
- Download: Download the capture as a .zip file. This option is only available if the capture completed successfully.
- Cancel: Stop an "In Progress" capture. A Capture Cancelled banner displays and an error will display in the log.
- Delete: Remove the capture files.

For a list of all network devices supported by BMC Atrium Discovery, see Supported SNMP devices.

**Viewing a subnet**

A subnet object in BMC Atrium Discovery represents a subnet, identified by an IP address range. Host systems are associated with a subnet. All subnet objects are set up automatically by the BMC Atrium Discovery process.

**To view a subnet**

1. Click the Subnets link in the Infrastructure Summary section of the Infrastructure page.
2. Select a subnet from the list.

   ![Subnet Example](image)

   **This screen illustrates a typical example of the Subnets page.**

   The information fields for a typical Subnet are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Range</td>
<td>Range of IP addresses.</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Interfaces discovered on the subnet. The headers can be used to sort the interface list.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>List of any missing fields.</td>
</tr>
<tr>
<td>Primary Data Provenance</td>
<td>Displayed only when showing provenance data.</td>
</tr>
</tbody>
</table>

**To run subnet reports**

A number of context-sensitive reports are available for subnets. See Subnet Reports for further information and examples of these subnet-related reports.
Viewing a software instance

A software instance object in BMC Atrium Discovery represents an instance of a software entity running in the environment. This often corresponds to one or more processes running on a host.

- To view a software instance
- What is Configipedia?
- To run software instance reports

To view a software instance

To view a software instance, click the Software Instance link in the Infrastructure Summary section of the Infrastructure page and select a Software Instance from the list.

The information fields for a typical Software Instance are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this software instance.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of software instance.</td>
</tr>
<tr>
<td>Configipedia Page</td>
<td>Associated Configipedia page link for this software instance. See #What is Configipedia? below.</td>
</tr>
<tr>
<td>Instance Count</td>
<td>Number of identical software instances.</td>
</tr>
<tr>
<td>Full Version</td>
<td>The version of the software plus the build number.</td>
</tr>
<tr>
<td>Product Version</td>
<td>The version of the software.</td>
</tr>
<tr>
<td>Host</td>
<td>The host on which this software is running.</td>
</tr>
<tr>
<td>Contains Software Components</td>
<td>Software components that this software instance contains.</td>
</tr>
<tr>
<td>Part of Application</td>
<td>The application that this software instance is a part of.</td>
</tr>
<tr>
<td>Details</td>
<td>Links to the details node for the software instance.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>The pattern used to create the software instance.</td>
</tr>
<tr>
<td>Primary processes</td>
<td>The primary process that identifies this software.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

What is Configipedia?

Configipedia is BMC’s community website that facilitates knowledge sharing around BMC Atrium Discovery software patterns, how they function and the real world issues they help address. Configipedia also provides visibility of the Technology Knowledge Update release schedule and contents.
A link to Configipedia is provided for Software Instance nodes and Business Application Instance nodes that are maintained by BMC Atrium Discovery software patterns. See Viewing an application instance. There is also a link to Configipedia provided in the detailed Pattern Module lists in the Knowledge Management section of the Discovery tab. See Viewing and editing a pattern module for more information.

To run software instance reports

A number of context-sensitive reports are available for software instances. See Software Instance Reports for further information and examples of these SI-related reports.

Viewing storage

Storage is discovered by BMC Atrium Discovery either via management software, or directly using SNMP queries.

- Where BMC Atrium Discovery discovers storage via management software, it creates an SI representing the management software.
- Where BMC Atrium Discovery discovers storage directly using SNMP, it creates a StorageDevice node representing the discovered storage device.

These events trigger the appropriate storage patterns to discover the associated storage components which are represented in the BMC Atrium Discovery UI and described in the following pages. Also provided is a brief guide to some of the terminology used in storage:

- Viewing a storage device
- Viewing a storage system
- Viewing a storage system group
- Viewing storage pools
- Viewing storage volumes
- Viewing storage processors
- Viewing storage connections
- Viewing disk drives
- Viewing front end FC ports
- Storage terminology

Viewing a storage device

A Storage device in BMC Atrium Discovery represents a storage chassis that has been discovered by a scan of its embedded WBEM/SMI-S provider.

To view a storage device

1. In the drop-down Search box, click Advanced Search. The Advanced Search box is displayed in the main frame.
2. Select Storage Devices in the Infrastructure panel.
3. Click Search.
3. If one storage device is found, then its page is displayed. If multiple storage devices are found, they are listed. Click the one that you want to view.

A typical example of a discovered storage device is illustrated in the following screen.

![Storage Device: tw_cluster-SAN](image)

The information fields for a discovered storage device are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the storage device. For example, tw_cluster-SAN.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The storage device vendor, for example, IBM.</td>
</tr>
<tr>
<td>Model</td>
<td>The model name, for example, IBM Storwize V3700.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>The storage device serial number, for example, 000000C02020230C.</td>
</tr>
<tr>
<td>OS Class</td>
<td>The operating system class, for example, Embedded.</td>
</tr>
<tr>
<td>OS Type</td>
<td>The operating system type, for example, IBM SVC Embedded SMI-S Provider. Or for NetApp devices, Data ONTAP®.</td>
</tr>
<tr>
<td>OS Version</td>
<td>The operating system version, for example, 7.1.0.36.</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>The operating system vendor, for example IBM.</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>A link to and details of the storage system that this storage device is part of. The following details are provided:</td>
</tr>
<tr>
<td></td>
<td>- Name</td>
</tr>
<tr>
<td></td>
<td>- Type</td>
</tr>
<tr>
<td></td>
<td>- Storage Access Mechanisms</td>
</tr>
<tr>
<td></td>
<td>- Vendor</td>
</tr>
<tr>
<td></td>
<td>- Model</td>
</tr>
<tr>
<td></td>
<td>- Serial Number</td>
</tr>
<tr>
<td></td>
<td>- World Wide Node Name</td>
</tr>
<tr>
<td></td>
<td>- Number of Pools</td>
</tr>
<tr>
<td></td>
<td>- Number of Volumes</td>
</tr>
</tbody>
</table>
### Field name | Details
--- | ---
| | - Number of Disk Drives
| | - Number of FC Ports
| | - Number of File Systems

#### Viewing a storage system

A Storage system in BMC Atrium Discovery represents the core chassis of a storage entity on the network on which servers can store data. A storage system is typically a rack mounted device with a large number of disks. Connectivity to the network is usually through a number of ports. Storage systems may enable clients to access the storage it is providing via a Storage Area Network (SAN) or Network Attached Storage (NAS).

#### To view a storage system

1. Click the **Infrastructure** tab, and then click **Storage Systems** in the Infrastructure Summary section.
2. Select a storage system from the list.

A typical example of a discovered storage system is illustrated in the following screen.

![Screen shot of a discovered storage system](image)

The information fields for a discovered storage system are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>The name of the storage system. For example, VSP@10.0.10.38.</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>The type of storage system. For example, Hitachi Virtual Storage Platform.</td>
</tr>
<tr>
<td><strong>Client Access Types</strong></td>
<td>The ways in which a client host can access the storage that this system is providing. This may be SAN, NAS, or SAN and NAS.</td>
</tr>
<tr>
<td><strong>Vendor</strong></td>
<td>The storage system vendor, for example, Hitachi.</td>
</tr>
<tr>
<td><strong>Model</strong></td>
<td>The model name, for example, VSP.</td>
</tr>
</tbody>
</table>
### Field name | Details
--- | ---
Serial Number | The storage system serial number, for example, 66534.
Total Disk Space | The total disk space managed by this storage system. See Storage terminology for more details.
Total Oversubscribed Capacity | The total oversubscribed capacity managed by this storage system. See Storage terminology for more details.
Storage Device | A link to the storage device through which the storage systems was discovered, if directly discovered.
Managed By | The software managing the storage system and the host on which it is running, for example, Hitachi HiCommand Server 7.4 on lon-box-01. This is also a link to the Software Instance page for that software. There may be more than one manager.
Storage Processors | A summary table showing the storage processors in the storage system. For each processor the following information is displayed:
- Processor Identifier
- Vendor
- Model
- Firmware
- Number of FC Ports
To see the storage processor page, click the corresponding table row. For more information, see Viewing storage processors.
Storage Pools | Displays the number of related storage pools. This is also a link to the Storage Pool List page. To see the storage pool page, click the corresponding table row. For more information, see Viewing storage pools.
Storage Volumes | Displays the number of related storage volumes. This is also a link to the Storage Volume List page. To see the storage volume page, click the corresponding table row. For more information, see Viewing storage volumes.
Disk Drives | Displays the number of related disk drives. This is also a link to the Disk Drive List page. To see the disk drive page, click the corresponding table row. For more information, see Viewing disk drives.

### Viewing a storage system group

A storage system group in BMC Atrium Discovery represents a high availability pairing of storage systems. Currently only NetApp storage systems are supported in this configuration.

#### To view a storage system group

1. In the drop-down Search box, click Advanced Search. The Advanced Search box is displayed in the main frame.
2. Select **Storage System Groups** in the **Infrastructure** panel.
3. Click **Search**.
   - If one storage system group is found, then its page is displayed.
   - If multiple storage system groups are found, they are listed. Click the one that you want to view.

A typical example of a discovered storage system group is illustrated in the following screen.
The information fields for a discovered storage system group are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the storage system group.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of storage system group. Currently this is NetApp.</td>
</tr>
<tr>
<td>Storage Systems</td>
<td>The storage systems that make up this group.</td>
</tr>
</tbody>
</table>

**Viewing storage pools**

The total number of storage pools associated with a storage device is displayed on the Storage System page. Click on the link corresponding to Storage Pools to view the Storage Pool List page. Click on a row to view the corresponding Storage Pool page. The following is a typical example of the Storage Pool page:

The Storage Pool page may display the following:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pool ID</td>
<td>The ID of the storage pool.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the storage pool. For example, Normal.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the storage pool. For example, NormalPool:1.0.</td>
</tr>
<tr>
<td>RAID Level</td>
<td>The RAID level configured for this pool. For example, RAID5(3D+1P)</td>
</tr>
<tr>
<td>Capacity</td>
<td>The usable capacity of the pool.</td>
</tr>
<tr>
<td>Consumed Capacity</td>
<td>The consumed capacity of the pool.</td>
</tr>
<tr>
<td>Percentage Consumed</td>
<td>The consumed capacity of the pool expressed as a percentage.</td>
</tr>
<tr>
<td>Available Capacity</td>
<td>The remaining capacity available.</td>
</tr>
</tbody>
</table>
### Field name | Details
---|---
Subscribed Capacity | The subscribed capacity of the pool expressed as a percentage.
Percentage Subscribed | The consumed capacity of the pool expressed as a percentage.
Oversubscribed Capacity | The oversubscribed capacity.
Oversubscription warning | Whether an oversubscription warning is active.
Storage System | The storage system for this pool. This is a link to the corresponding Storage System page.
Storage Volumes | The volumes for the storage system. This is a link to the corresponding Storage Volume page.

See [Storage terminology](#) for more details of terms such as Capacity, Consumed capacity, Oversubscription and so on.

### Viewing storage volumes

The total number of storage volumes associated with a storage device is displayed on the Storage System page. Click on the link corresponding to Storage Volumes to view the Storage Volume List page. The following is a typical example of the Storage Volume page:

![Storage Volume page](image)

The Storage Volume page may display the following:

### Field name | Details
---|---
Volume ID | The ID of the storage volume. For example, HITACHI R500D7A50000.
Name | The name of the storage volume.
NAA ID | The NAA ID. For example, 60:05:07:63:00:80:80:8c:30:00:00:00:00:00:00:06.
<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapped</td>
<td>Whether or not the volume is mapped.</td>
</tr>
<tr>
<td>Thinly Provisioned</td>
<td>Displays whether the volumes are thinly provisioned or not. For example, if the volumes are not thinly provisioned, it displays <code>False</code>.</td>
</tr>
<tr>
<td>Visible Capacity</td>
<td>The visible capacity</td>
</tr>
<tr>
<td>Consumed From Pool</td>
<td>The capacity consumed from the pool.</td>
</tr>
<tr>
<td>Consumable Capacity</td>
<td>The consumable capacity.</td>
</tr>
<tr>
<td>Percentage of Pool Consumed</td>
<td>The percentage of the pool consumed.</td>
</tr>
<tr>
<td>Block Size</td>
<td>The block size of the storage volume.</td>
</tr>
<tr>
<td>Number of Blocks</td>
<td>The number of blocks of the storage volume. For example, 5066268672.</td>
</tr>
<tr>
<td>Consumable Blocks</td>
<td>The number of consumable blocks of the storage volume. For example, 5066268672.</td>
</tr>
<tr>
<td>Volume Size</td>
<td>The size of the storage volume.</td>
</tr>
<tr>
<td>File System</td>
<td>The file system that the host is using to store files, shown as it is mounted from that host.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>The virtual disks used to hold the file system.</td>
</tr>
<tr>
<td>Storage System</td>
<td>The storage system for this volume. This is a link to the corresponding Storage System page.</td>
</tr>
<tr>
<td>Storage Pool</td>
<td>Displays the following fields about the storage pool and each field link to the corresponding storage pool page:</td>
</tr>
<tr>
<td></td>
<td>• Pool ID</td>
</tr>
<tr>
<td></td>
<td>• Type</td>
</tr>
<tr>
<td></td>
<td>• Name</td>
</tr>
<tr>
<td></td>
<td>• Capacity</td>
</tr>
<tr>
<td></td>
<td>• Consumed Capacity</td>
</tr>
<tr>
<td></td>
<td>• Percentage Consumed</td>
</tr>
<tr>
<td></td>
<td>• Available Capacity</td>
</tr>
<tr>
<td></td>
<td>• Subscribed Capacity</td>
</tr>
<tr>
<td></td>
<td>• Percentage Subscribed</td>
</tr>
<tr>
<td></td>
<td>• Oversubscribed Capacity</td>
</tr>
<tr>
<td></td>
<td>• Oversubscription Warning</td>
</tr>
<tr>
<td></td>
<td>• Number of Volumes</td>
</tr>
<tr>
<td>Storage Consumers</td>
<td>The consumers of the storage volume. The following fields may be displayed:</td>
</tr>
<tr>
<td></td>
<td>• Connection Name: The name of the connection.</td>
</tr>
<tr>
<td></td>
<td>• Connection Type: Type of the connection.</td>
</tr>
<tr>
<td></td>
<td>• Connection Value: The connection value.</td>
</tr>
<tr>
<td></td>
<td>• Fibre Channel Port: The connection name.</td>
</tr>
<tr>
<td>Front End FC Ports</td>
<td>The following front end fibre channel port fields may be displayed, which also links to the corresponding Fibre Channel Port page:</td>
</tr>
<tr>
<td></td>
<td>• Port Name: The front end fibre channel port name.</td>
</tr>
<tr>
<td></td>
<td>• WWPN: The World Wide Port Name.</td>
</tr>
<tr>
<td></td>
<td>• Type: The front end fibre channel port type.</td>
</tr>
<tr>
<td></td>
<td>• State: The front end fibre channel port state.</td>
</tr>
<tr>
<td></td>
<td>• Speed: The front end fibre channel port speed.</td>
</tr>
</tbody>
</table>
**Io Group**  
The IOP group.

**Virtualization Type**  
The virtualization type. For example, Sequential.

**Compressed**  
Whether the volume is compressed.

### Viewing Storage Processors

The storage processors associated with a storage device are displayed on the Storage Processors section of the Storage System page. The following is a typical example of the Storage Processor page:

---

**Field name** | **Details**
---|---
**Processor Identifier** | The processor identifier. For example, VSP.55205.0.
**Vendor** | The hardware vendor. For example, Hitachi.
**Model** | The model name. For example, VSP.
**Description** | The model description.
**Serial** | The serial number.
**Firmware** | The firmware details.
**Storage System** | The name of the storage device. It is a link to the corresponding Storage System page.
**Front End FC Ports** | The following fibre channel front-end port fields may be displayed (which are also links to the corresponding Fibre Channel front end ports page):

- **Port Name**: The fibre channel front-end port name.
- **WWPN**: The World Wide Port Name.
- **Type**: The fibre channel front-end port type.
- **State**: The fibre channel front-end port state.
- **Speed**: The fibre channel front-end port speed.
Viewing storage connections

A Storage connection represents a connection between a storage client (a host using the storage) and a storage volume. The same connection may be linked to multiple volumes representing the client's visibility of multiple storage volumes. The volumes shown are only those that the client is permitted to access.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection Name</td>
<td>The name of the connection. For example: 10000000c96eb8db/10000000c96eb8db</td>
</tr>
<tr>
<td>Connection Type</td>
<td>The type of storage connection. This can be WWPN or iSCSI IQN. At pre-release 2, only WWPN connections support linking to the host using storage.</td>
</tr>
<tr>
<td>Connection Value</td>
<td>A unique identifier for the storage consumer using this connection.</td>
</tr>
<tr>
<td></td>
<td>• For WWPN connections this is the fiber channel port name of the host, for example:10:00:00:00:c9:6e:b8:db.</td>
</tr>
<tr>
<td></td>
<td>• For iSCSI IQN connections this is iSCSI qualified name (IQN).</td>
</tr>
<tr>
<td>Storage Volumes</td>
<td>A link to any storage volumes visible to the host using this connection.</td>
</tr>
<tr>
<td>Consumer Fibre Channel Port</td>
<td>Information on the Fiber channel port on the host using this connection.</td>
</tr>
<tr>
<td></td>
<td>• Port Name</td>
</tr>
<tr>
<td></td>
<td>• WWPN</td>
</tr>
<tr>
<td></td>
<td>• Type</td>
</tr>
<tr>
<td></td>
<td>• State</td>
</tr>
<tr>
<td></td>
<td>• Speed</td>
</tr>
</tbody>
</table>

Viewing disk drives

The total number of disk drives associated with a storage entry is displayed on the Storage System page. The following is a typical example of the Disk Drive page:
The Disk Drive page may display the following:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the disk drive.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the disk drive.</td>
</tr>
<tr>
<td>Speed</td>
<td>The speed of the disk drive in RPM.</td>
</tr>
<tr>
<td>Size</td>
<td>The size of the disk drive in bytes.</td>
</tr>
<tr>
<td>Vendor</td>
<td>The disk drive vendor. For example, Hitachi.</td>
</tr>
<tr>
<td>Model</td>
<td>The disk drive model. For example, DKS5D-J900SS.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>The disk drive serial number.</td>
</tr>
<tr>
<td>Role</td>
<td>The role of the disk. For example, raid.</td>
</tr>
<tr>
<td>Current State</td>
<td>The current state of the disk drive. For example, OK.</td>
</tr>
<tr>
<td>Virtual</td>
<td>A flag stating whether the disk is virtual.</td>
</tr>
<tr>
<td>Storage System</td>
<td>The name of the storage device. It is a link to the corresponding Storage System page.</td>
</tr>
<tr>
<td>Host</td>
<td>Host containing this disk.</td>
</tr>
<tr>
<td>File Systems</td>
<td>File systems using this disk.</td>
</tr>
<tr>
<td>Physical Disks</td>
<td>Physical disk for this virtual disk.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>Virtual disks using this physical disk.</td>
</tr>
<tr>
<td>Physical Volumes</td>
<td>Physical storage volume for this virtual disk.</td>
</tr>
</tbody>
</table>

Viewing front end FC ports

The front end fibre channel (FC) ports associated with a storage processor are listed on the Front End FC Ports section of the Storage Processor page. The following is a typical example of the Fibre Channel Port page:
The following fields may be displayed:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port Name</td>
<td>The name of the front end FC port.</td>
</tr>
<tr>
<td>WWPN</td>
<td>The World Wide Port Name.</td>
</tr>
<tr>
<td>Role</td>
<td>The role of the fibre channel port. For example, Front End.</td>
</tr>
<tr>
<td>Type</td>
<td>The front end FC port type. For example, Public Loop (NL)</td>
</tr>
<tr>
<td>State</td>
<td>The front end FC port state. For example, Operational.</td>
</tr>
<tr>
<td>Speed</td>
<td>The front end FC port speed.</td>
</tr>
<tr>
<td>Storage Processor</td>
<td>The storage processor for this port. This is also a link to the Storage Processor page.</td>
</tr>
<tr>
<td>Exposed Volumes</td>
<td>The volumes exposed through this port. This is also a link to the Storage Volume List page.</td>
</tr>
</tbody>
</table>

**Storage terminology**

This section provides a brief guide to some of the terminology used in storage.

**Provisioning**

Provisioning refers to the availability and allocation of the physical storage (or blocks on disk).

**Traditional provisioning:** Physical storage is allocated and available to consuming hosts at the time the storage is set up. Also called "fat" or "thick" provisioning.

**Thin provisioning:** Physical storage is allocated on demand. Unused storage is available but it has not been allocated. Thin provisioning is designed to avoid the hardware and energy costs incurred where pre-allocated storage remains unused. Enables storage to be *over-subscribed*.

**Size and capacity**

Size and capacity refer to the amount of data that a storage entity can store.

**Size:** The underlying size of the device, the number of bytes of data that can fit on the device.

**Capacity:** The amount that can actually be used.

For example, if a storage entity has two 1 TB disks in a RAID 1 pair, the size of the storage is 2 TB, but the capacity is 1 TB.
Volume capacities

Visible capacity: The visible capacity is the amount of storage that is available to the consuming host.

Consumed capacity: The amount of capacity that the consuming host has used.

- With traditional provisioning the consumed capacity equals the visible capacity. The physical storage was allocated before it is used, at the time the storage was set up.
- For thin provisioning this is less than the visible capacity. The consuming host can see the full amount of storage available to them even though some of it is not allocated.

Pool capacities

Capacity: The total amount of storage that is in the pool.

Consumed capacity: The amount of capacity that has been used by the volumes and subsidiary pools allocated from this pool.

Available capacity: The unused storage in the pool. The capacity minus the consumed capacity.

Subscribed capacity

Subscribed capacity: The amount of capacity that can be used by volumes and subsidiary pools allocated from this pool. With thin provisioning the subscribed capacity can be greater than the pool capacity. This is described as over-subscribed.

Over-subscribed capacity: The subscribed capacity minus the capacity.

Over-subscription warning: A warning flag set if the consumed capacity of an over-subscribed storage entity reaches 75%.

This diagram shows a graphical representation of subscribed capacity, over-subscribed capacity, and the over-subscription warning.
Reporting on the infrastructure

This topic provides instructions for running reports on infrastructure items.

To run standard reports on infrastructure items

1. Click the Infrastructure tab in the Primary Navigation Bar.
2. On the Infrastructure page, click any report name to run the report.
   Some reports are run immediately. For those that take parameters, an intermediate screen is displayed.
3. If an intermediate screen is displayed, define appropriate selection criteria and click Run.
   The results are summarized in the form of a list page.
4. To display the View Object page of any of the objects found, click the relevant entry.

See Infrastructure reports for further information and details of the infrastructure-related reports.

Managing your business applications

The Applications section of BMC Atrium Discovery enables you to view and search details of your business software, including information about the software applications that are actually running on your system.

- Viewing details of business applications
- Searching for application information
- Viewing an application instance
- Reporting on business applications
Viewing details of business applications

The Applications page of BMC Atrium Discovery enables you to view and search details of your business software, including information about the software applications that are actually running on your system. The Applications page includes Business Application Instances (BAIs). Like Software Instances (SIs), BAIs are determined by patterns.

On the main page of the Applications tab you can view a summary of:

- All the application items held in the system
- A number of application-related reports

From any list of objects you can drill down to display more detailed information, such as the details of a host or an SI and the relationships between objects.

- To view a summary of the business applications items
- To view a list of business items
- To view an application in detail
- Creating a manual group for application mapping

To view a summary of the business applications items

Click the Applications tab in the primary navigation bar. The Applications page displays the total number of application instances in the system, provides an application mapping channel, and displays a list of application-related reports. See Application reports.

This screen illustrates the summary information for business application items on the Applications page.

To view a list of business items

Click the Application Instances link.
To view an application in detail

1. Display an application list page, for example:
   - The results of any search displaying applications.
   - Any report listing applications.
2. Click any Business Application in the list to display the View Object page.
   The page lists all the attributes of the object and its relationships with other objects.
3. View any other related object, or delete the object.

Some attributes or relationships appear on the View Object page only if a data value has been set.

Creating a manual group for application mapping

The Application Mapping channel enables you to create a manual group to use in application mapping and provides a list of links to existing manual groups.

To create a manual group

1. In the Application Mapping channel, enter the name of the new group in the Create group for mapping field. The period (.) double quote ("), and dollar ($), symbols cannot be used in a group name.
2. Click Create.
   The Home Page for the new empty group is displayed.

   ![This screen illustrates the Home Page for a new group.]

To add a subgroup to a group

1. Select the nodes to add to the new subgroup.
   You can use the select tools to help with this: (Select: All | None | Invert).
2. Click the Actions button to display the Actions menu, and select the Manual Groups item.
3. If the required parent group already exists, click the entry in the displayed list to enter its name in the Group field (or enter the name of the group manually).
   After the name is entered, the non-matching groups are removed from the displayed list and the Subgroup entry field is enabled.
4. If the parent group does not already exist, create it by entering a name in the Group field.
5. Enter the name of the subgroup into the Subgroup entry field.
This screen illustrates how to add a subgroup to a group on the Home Page.

6. Click **Create**.
   You can add an empty subgroup from the home page for the group.

To view a group page

Click the name of the group in the group list.
The Home page for the group displays the **Application Mapping** tab.

This screen illustrates how to view a group on the Home Page.

**Searching for application information**

You can search for information in the BMC Atrium Discovery datastore by clicking the **Search Options** icon. This displays a drop-down dialog where you can tailor your searches on all pages in BMC Atrium Discovery.

To run a basic application search

1. In the **Search Options** drop-down dialog, enter the keyword or text string you want to search for.
2. Choose **Applications** from the list.
3. To search for an entire attribute, select **Exact Match**.
4. To search for an entire word, select **Word Match**.
   The **Show Data Completeness** check box indicates whether or not Data Completeness is shown for each returned object. Select this check box if you need to view the data quality for all items.
5. Click **Search**.
   Summary details of objects matching your criteria are listed in the main frame. If your search returns too many matches, you can refine it by running an additional search on the items.
Related topics

Searching for data.

Viewing an application instance

An application instance is a running instance of a business application; an identifiable application that supports a particular business function. All application instance objects are set up automatically by the BMC Atrium Discovery Discovery and Reasoning process.

To view an application instance

1. Click the Application Instances link in the Application Summary section of the Applications page.
2. From the Application Instances page, select an application instance from the list.

The information fields for a typical Application Instance are detailed in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the application.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of application instance.</td>
</tr>
<tr>
<td>Configipedia Page</td>
<td>The page in Configipedia that describes the application.</td>
</tr>
<tr>
<td>Full Version</td>
<td>The full version number of the application.</td>
</tr>
<tr>
<td>Build</td>
<td>The build number of the application.</td>
</tr>
<tr>
<td>Product Version</td>
<td>The product version of the application.</td>
</tr>
<tr>
<td>Contains Software Instances</td>
<td>Software instances that are part of this application instance.</td>
</tr>
<tr>
<td>Hosts</td>
<td>Hosts that this application is running on.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Pattern responsible for creating and maintaining this BAI.</td>
</tr>
<tr>
<td>Data Completeness Issues</td>
<td>List of any missing fields.</td>
</tr>
</tbody>
</table>

Reporting on business applications

This topic provides instructions for running standard reports on business applications.

To run standard reports on business applications

1. Click the Application tab.
2. Click any report name to run the report.
   Some reports are run immediately. For those that take parameters, an intermediate screen is displayed.
3. If an intermediate screen is displayed, define appropriate selection criteria and click Run.
   The results are summarized in the form of a list page.
4. To display the View Object page of any of the objects found, click the relevant entry.

Related topics
Application reports

Auditing a data center

A data center baseline is an audit performed before any large IT project. The extent of the baseline, the inventory (hardware, software, physical, virtual), and the inter-dependencies must be considered. They are typically difficult, time consuming, and rapidly become outdated if not automated. BMC Atrium Discovery makes them a lot simpler.

- Automatic grouping
- Manual grouping
- Creating a host profile

What is baselining

No large IT project, especially one that will affect hundreds or even thousands of servers, can be started without a baseline. Such baselines are also referred to as IT audits. Virtualization and consolidation projects also require an understanding of the performance of the infrastructure: how busy the hosts are, how much power and cooling they require and if there are any reasons why they should not be virtualized or consolidated.

Items that are typically audited

generally involve an in-depth investigation into the IT infrastructure to discover the following:

- Its extent: Are there hosts that you're not aware of? The answer is invariably "yes". What are they doing?
- Inventory: What hardware and software are the hosts running? Are they physical or virtual? Where are they located?
- Dependencies: This is the longest, most complicated and expensive part of the baseline.
  - Which hosts and software provide which services?
  - Which IT staff own and operate the hosts?
  - Which business services (business applications) are provided by which infrastructure?
  - Which business units rely on which business applications?
  - What are the interdependencies between hosts, and therefore between business applications?
Where BMC Atrium Discovery can help

BMC Atrium Discovery gathers essential hardware and software information quickly and accurately.

BMC Atrium Discovery's Baseline Dashboard tells you the progress of your baseline at a glance — how much of the estate you have gathered information about, what the spread of operating systems is on the hosts discovered so far, how many are virtual, graphs of the progress you are making, and more. You can display information on a Baseline Dashboard to help you baseline your project, such as a data center consolidation. Additionally, BMC Atrium Discovery helps you determine the relationships and dependencies in your estate only hours after deployment.

Client workshops, interviews and ad-hoc conversations with technical staff are the most common techniques for finding the business dependencies in and on your data center. Before taking up these people’s valuable time, you must know which questions to ask to whom. You have to have a lot of detailed starter information about how the estate is structured: which hosts are communicating with which others (with a rough idea of why), which hosts seem to be working closely together, which hosts seem to be relied on by a large proportion of the estate and so forth.

Normally, this initial analysis is done manually by examining lists of processes, the network communications on which the processes are involved in, and the host communications for hundreds or thousands of hosts, which can be a tedious, error-prone exercise that can take considerable time.

After the initial analysis is complete, the workshops are conducted. These involve talking to on-site experts about your conclusions from the analysis. These conclusions will often be challenged; either your expert is aware of a subtlety that you hadn't picked up on, or the environment has moved on a bit and they're not aware of some new condition. Either way, you need a way to present your conclusions to them with clear supporting evidence that can be discussed and further analysed. Frantically cross-referencing reams of reports under the gaze of an impatient technical architect wastes time and precious good-will.

Automatic grouping

When you baseline a data center, one of the first tasks is to subdivide it into small groups of hosts. BMC Atrium Discovery does this automatically for all discovered hosts at the end of each scan.

Automatic grouping is intended as a baselining tool for use at the scale of 100 to 500 hosts. At larger scale, the usefulness and usability of the visualizations is diminished.

- How automatic grouping works
- To see how your hosts are grouped
- To interact with the visualizations
- To observe host communication
- To exclude hosts from automatic grouping
How automatic grouping works

There are three types of group created automatically:

- Group around a host
- Clients of a host
- Hosts with no relevant communication

When a host is involved in automatic grouping, the Host:Host:ObservedCommunication: Host:Host relationship (Communicating With) is displayed on the host page.

An additional group type can be created manually by excluding hosts from automatic grouping. These are placed into a group called "Hosts Excluded from Automatic Grouping". This is described in Excluding hosts from automatic grouping below.

You can enable or disable automatic grouping in the Discovery configuration page.

These groups are created in the following way after each scan:

- Groups hosts which communicate with one another — These are called Group around hostname.
  Any two hosts which are seen to communicate are called neighbors. When a pair of hosts are considered, the connection between them is ranked. A higher rank increases the probability of them being grouped. The ranking is calculated according to the proportion of their total neighbors that they share. Where more neighbors are shared, the ranking is higher, and the hosts are more likely to be grouped. This is used to reduce the chances of clients of shared infrastructure being grouped on the basis of those connections.
- Groups hosts which are clients of a particular set of servers — These are called Clients of host.
- Hosts with no observed communication — These are grouped in the "Hosts with no relevant communication" group. Further hosts, that communicate only with hosts that have been excluded from grouping, are also placed in this group.

Host Groups are stored as AutomaticGroup nodes in the datastore.

To see how your hosts are grouped

There are several ways to view automatic grouping.

Automatic grouping channel

The automatic grouping channel shows a summary of the current grouping results:
This screen illustrates the automatic grouping channel.
Click any of the icons to view a visualization. The top row shows:

- The overview. This is a visualization that shows all automatic groups and how they communicate with each other.
- No relevant communication. A visualization of the group containing hosts that have either no communication at all or are communicating only with excluded hosts.
- Excluded hosts. Hosts that have been excluded from the automatic grouping process (see below).

The following line shows up to five of the most highly connected groups; that is, those that are communicating with the most other groups. The final line shows up to five of the largest unconnected groups, self-contained groups that do not communicate outside themselves at all.

Direct navigation

1. Select the Infrastructure section from the Primary Navigation bar.
2. From the Infrastructure page, click the Explore tab on the Dynamic Secondary Navigation bar.

To interact with the visualizations

The controls (Zoom & Pan, Layout, Printing & Export) operate in the same way as Dependency Visualizations. The usual menu options also appear when clicking on a node (View Details, Manual Groups..., Hide, and Unhide Neighbors, although there is no Re-Center option) and clicking on the background (Unhide All).

Right clicking on an Automatic Group displays a pop-up menu with the usual four menu options listed above and an additional option, Open, which shows a more detailed visualization of the clicked group. This is a drill-down option, and shows the hosts that compose the automatic group, how they are communicating, and the other automatic groups they are communicating with.
This screen illustrates an example of the display for a drill-down option for automatic grouping.

Host nodes show an additional menu option that controls whether the host is included in automatic grouping. The menu item shows as either **Exclude From Automatic Grouping** or **Include In Automatic Grouping**, depending on the current state of the node. See below for more information about this.

The Automatic Grouping palette block on the left has a link to return to the overview visualization from any detail visualization.

**To observe host communication**

To see details of why two hosts are communicating, right-click the edge between them, and choose the only menu option, Show Communication Details.

A report of all directly discovered network connections that contributed to the observed communication link between the hosts is displayed.
This screen illustrates a report of all directly discovered network connections contributing to an observed communication link between hosts.

To exclude hosts from automatic grouping

Some hosts in your estate might be *super-connected* which means they are connected to many, perhaps all other hosts in the estate. This might be interesting information. However, once noted, it can be useful to remove these hosts from the grouping algorithm in order to focus on the other hosts. BMC Atrium Discovery lets you do this by marking such hosts to be excluded from automatic grouping by using the **Exclude From Automatic Grouping** option.

After a host is marked as excluded from automatic grouping, it is displayed differently to make this obvious: it is grayed out slightly, and has an excluded icon overlayed on it.

This screen illustrates a host marked excluded from automatic grouping.

To actually see the results of excluding the host, the grouping algorithm must be re-run. Several hosts can be marked as excluded from automatic grouping before re-running the algorithm. To re-run the algorithm, click **Update** that appears in the Automatic Grouping palette block on the left. Regrouping then occurs and the overview visualization is redisplayed.
Hosts that are excluded from automatic grouping appear in a special group, labeled Hosts Excluded From Automatic Grouping. This is displayed on the overview visualization in a similar way to hosts with no observed communication.

This screen illustrates an example of hosts that are excluded from automatic grouping appearing in a special group.

Drilling into this group shows all the hosts in the group, how they are communicating, and the other groups with which they are communicating, as before. However, hosts in the other groups with which the excluded hosts are communicating no longer show a communicating relationship to the excluded host, and the groups are calculated without regard for the communications to and from the excluded hosts. This also means that hosts that are communicating only with excluded hosts are moved into the Hosts with no relevant communication group.

Hosts can be included again by right-clicking an excluded host and choosing **Include In Automatic Grouping**.

**Manual grouping**

BMC Atrium Discovery enables you to add manually selected nodes to a group. You can use this feature to create new groups, or to modify **automatically created** groups. Manually created groups can overlap, and can contain some or all of the nodes in automatically created groups. Manually created groups can also contain multiple node kinds (for example, hosts, Software Instances, and Business Application Instances) and subgroups. Groups and subgroups enable you to collect all items which make up an application and divide them into subgroups to help you understand the structure and operation of an application. The scheme that you use to subdivide the components of the application is the basis for **understanding the application**, it does not affect the way the applications are mapped.

⚠️ **Non-ASCII Unicode characters in CAM**
In Collaborative Application Mapping (CAM), if you create components such as group names or functional components, that contain non-ASCII Unicode characters, the Business Application Instance (BAI) that results from running the pattern displays with unreadable characters.

After you have collected the items and divided them into subgroups, you can create a PDF report to distribute to the application or component owners.

The Group page is the home page for each group that has been created in BMC Atrium Discovery.

This screen illustrates the home page for an example manual group.

You can access Group pages from the Groups menu.

The following topics provide instructions and information for using manual grouping:

- Viewing created groups
- Adding nodes to an empty group
- Creating manual groups
- Managing manual groups
- Reporting on manual groups

**Viewing created groups**

The Group page is the home page for each group that has been created in BMC Atrium Discovery. The default view is the Contents tab.

The Group page lists the node types in the group and displays the first few nodes of each type. Click the node type heading (one of Hosts, Application Instance, or Software Instances) to view a list view of nodes of that type in the group. You can also click a node name to see the node view page for that node.

This screen illustrates the group home page.
The top of the page includes a Notes section where you can add and edit notes on the group. The notes can be typed in rich text format, and you can include paragraph formatting, HTML links, and so forth. By clicking Edit, you can go back and revise the text. For more information about the notes functionality, see Searching and investigating the datastore.

You can also create PDF reports on the group and individual subgroups. For more information, see Reporting on manual groups.

The following screen shows a group with subgroups.

This screen illustrates the group page showing populated groups and subgroups.

Adding nodes to an empty group

When you create a new group from the Application Mapping channel, the Home Page for the new empty group is displayed.

In an empty group, the following label is displayed "This Group is empty, please browse and add items. You could start by trying an Advanced Search." Part of this is a link to the Advanced Search page.

For example, if the application that you are mapping has components with a specific naming scheme, start by searching for Software Instances which include a fragment of that name. For example, in the basic example to map BMC Atrium Discovery, enter "BMC Atrium Discovery" in the keywords field and click Search. Select Software Instances from the resulting multi-kind list and from the resulting list select components which are likely to be part of the application, in this case, select all of the software instances. Using the Query Builder you can now traverse along relationships to find the hosts running those Software Instances and add them to the group.
Common examples (depending on your application) of other related node types that you might consider traversing to and adding to your groups are:

- Software Components
- Database Details
- SIs with this Detail (from Database Details)
- Software Instances representing the database server

Once you have added more nodes, you should look at the group again and consider items that might have been missed. To simplify managing large groups, you might also wish to divide the group into a number of subgroups. There are a number of ways you can choose to do this, depending on the application that you are mapping. Examples are:

- Application layers
- Ownership
- Geography
- UAT/Production
- Unknown components that require investigation

You can reorder the subgroups by dragging and dropping them in the Contents window.

Creating manual groups

You can create manual groups from a node page, a search result page, or a visualization.

- To create a manual group from a node view or search result page
- To create a manual group from a visualization

To create a manual group from a node view or search result page

1. Click the Groups icon to display the Groups menu.

   ![Manual Groups]

   Although this step is not necessary, it makes the operations clearer in the UI.
3. Click away from the menu to close it.
4. If you are on a search result page, select the nodes to add to the new group.
   You can use the select tools to help with this: (Select: All | None | Invert).
5. Click Actions to display the Actions menu, and select the Manual Groups item.
6. Enter the name of the new group in the field on the Groups page, as shown in the following screen.

   ![Note]
   
   Double quotes (") cannot be used in a Group name.

7. Click Create.
   The page is updated to show that the node or nodes have been added to the new group.

To add a subgroup to a group

1. Select the nodes to add to the new subgroup.
   You can use the select tools to help with this: (Select: All | None | Invert).
2. Click Actions to display the Actions menu, and select the Manual Groups item.
3. If the required parent group already exists, click the entry in the displayed list to enter its
   name in the Group field (or enter the name of the group manually).
   After the name is entered, the non-matching groups are removed from the displayed list and
   the Subgroup entry field is enabled. If the parent group does not already exist, create it by
   entering a name in the Group field.
4. Enter the name of the subgroup into the Subgroup entry field.

   ![This screen illustrates how to add a subgroup to a group on the Home Page.]

5. Click Create.

You can add an empty subgroup from the home page for the group.

To create a manual group from a visualization

1. Click the Groups icon to display the Groups menu.
2. From the Groups menu, select Show group labels in visualizations.
   Although this step is not necessary, it makes the operations clearer in the UI.
3. Click away from the menu to close it.
4. Select the nodes to add to the new group.
5. Right-click one of the selected nodes and select Manual Groups from the menu.
6. Enter the name of the new group in the field on the Groups page.

![Warning]

**Note**

Double quotes (") cannot be used in the Group name.

7. Click **Create**.
The visualization is refreshed, and the members of the group are highlighted with a tag showing the group name.

To add a subgroup from a visualization

1. Select the nodes to add to the new subgroup.
2. Right-click one of the selected nodes and select Manual Groups from the menu.
3. If the required parent group already exists, click the entry in the displayed list to enter its name in the Group field (or enter the name of the group manually).
   After the name is entered, the non-matching groups are removed from the displayed list and the Subgroup entry field is enabled. If the parent group does not already exist, create it by entering a name in the Group field.
4. Enter the name of the subgroup into the Subgroup entry field.
5. Click **Create**.
The visualization is refreshed, and the members of the subgroup are highlighted with a tag showing the subgroup name.

### Managing manual groups

This topics explains how to manage manual groups.

- To view a manual group by using the Groups menu
- To view the contents of a group by using a generic query
- To add a node to a group
- To remove a node from a group
- To rename a group
To use a working set
To delete a manual group

To view a manual group by using the Groups menu

1. Click the **Groups** icon to show the **Groups** menu.
2. Select the name of the group to view.

![This screen illustrates the home page for the selected group.](image)

See the **group home page** for more information.

To view the contents of a group by using a generic query

1. Click the **Search** icon to the left of the Search box at the top right of the UI.
2. Click the **Generic Search Query** link.
3. Enter the query in the text entry field.
   
   For example, to display the contents of the Group AR host Pair, enter the following query and click **Run Query**:
   
   ```
   search Group where name="AR host Pair" 
   traverse Container:Container:ContainedItem: 
   Containment:ContainedItem: 
   
   The results are displayed in a report.
   ```

   To display the contents of subgroups, you need an additional containment traversal, for example:

   ```
   search Group where name="AR host Pair" 
   traverse Container:Container:ContainedItem: 
   traverse Container:Container:ContainedItem: 
   ```

   For more information on running queries, see **Using the Search service**.

To add a node to a group

You can add nodes of any type to an existing group, irrespective of the current contents of that group. From a node view page, search result page, or visualization:
If on a search result page or visualization, select the node or nodes that you would like to add to the group.

If on a visualization, right click one of the selected nodes, otherwise click **Actions** and select the Manual Groups menu item.

If using the selection check box, select the group or groups to which to add the nodes. The nodes are added immediately.

If you select a number of hosts where only some are in a group, the selection check box shows a third state which means that not all selected hosts are members of that group. In the following example, only one of the two selected hosts is in the Red Hat group.

This screen illustrates only one of the two selected hosts in the Red Hat group.

After you have created a group, click its entry in the groups menu to visit the home page for that group.

**To remove a node from a group**

You can remove nodes from a node view page, search result page, or visualization.

- If on a search result page or visualization, select the node or nodes that you would like to add to the group.
- If on a visualization, right click one of the selected nodes, otherwise click **Actions** and select the Manual Groups menu item.
- If using the selection check box, select the group from which to remove the nodes. The nodes are removed immediately.

The **Groups** menu enables you to view, rename, and delete manually created groups, select a group or groups to be a working set. To the working set, you can restrict viewing, and configure the display of labels in lists and visualizations.
This screen illustrates the Groups menu.

To rename a group

To rename a group, click the pencil icon that corresponds to the group on the Groups menu. The name is redisplayed in an editable field. Rename the field as required and click Enter to apply the change. Double quotes (" defense) cannot be used in the group name.

To use a working set

A working set is a group or groups that you have selected to work with. You can tag any group as part of your working set. Working sets are per user and are not shared. Any group that you create automatically becomes part of your working set. Groups in your working set are denoted by a yellow star. You manage working sets by using the Groups drop-down in the dynamic toolbox or the groups dialog that is accessible from the Actions menu or the Visualization node menu. To add a group to your working set, click the gray star icon. This changes to the yellow star icon signifying that it is now in your working set. Clicking the yellow star icon removes the working set status and the icon reverts to the gray star icon.

To delete a manual group

Select the Delete icon from the Groups menu. Confirmation is required unless the group is empty.

⚠️ Note

Deleting a group does not delete the nodes contained in that group. Only the group itself is removed.
Reporting on manual groups

After you create groups, you can create a report from the Group page that details information about the group, including Configuration Items (CIs), subgroups, and any multiple node kinds. This report contains the output of three subgroups, all with the same content. You can create the report with different levels of detail, each that helps you preview the information to collaborate on application mapping prototypes.

The PDF displays one page per functional component, and a tree view of the functional component can also be used as an index.

A visualization is shown of the entire group. This shows how the items contained in a functional group are linked to their host or MFPart directly or through intermediate nodes, once the pattern for the application has been generated and run. This can be used as a quick preview as you are mapping the application.

To create group reports

1. On the Group page, click the Contents tab.
2. Select the level of detail you want in the report by using one of the following methods:
   a. Click an area on the Reporting Level bar, where the lowest level is located on the left and the highest level is located on the right (Medium is the default setting), as illustrated in the following screen.

   ![Reporting Level Bar Illustration]

   This screen illustrates how to adjust the reporting level of the group PDF report.

   b. Choose one of the following reporting levels from the Actions menu:

<pre><code>  | Level | Type of report generated |
  |-------|--------------------------|
  | Set PDF Detail High | Details the names of the nodes in each group and a list of attributes for each node, and displays functional components with all relationships to other components. |
  | Set PDF Detail Medium | Details the names of the nodes in each group and a list of important attributes for each node. This is the default reporting level. |
</code></pre>
<table>
<thead>
<tr>
<th>Level</th>
<th>Type of report generated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set PDF Detail Low</td>
<td>Details the names of the nodes in each group, but no further information on the nodes. You might choose this level, for example, if you have complex diagrams in your report where the links or relationships are not readable. This would display only the functional components and not their relationships to other components.</td>
</tr>
</tbody>
</table>

3. Choose **Generate PDF Report** from the **Actions** menu.
4. From the status link at the top of the Group page, click the **Reports Tab** link.
5. Select **Group Report** in the list and click **Download**.

The following example shows all three levels combined in one report to represent a three-tier application here.

### Creating a host profile

The host profile is a PDF report that provides a summary of baselining information about a host or hosts.

- To create a host profile for a single host
- To create a host profile for selected hosts or all hosts
- To view existing host profiles

A sample host profile is available here.

**To create a host profile for a single host**

1. Click the **Hosts** link in the Infrastructure Summary section of the Infrastructure page.
2. Select a host from the list.
3. From the **Actions** menu, choose **Create Host Profiles** to generate and download the host profile for the selected host.

**To create a host profile for selected hosts or all hosts**

1. Click the **Hosts** link in the Infrastructure Summary section of the Infrastructure page.
2. Select the hosts that you want to report on.
3. From the **Actions** menu, choose **Create Host Profiles** to generate and download the host profile for the host.

**To view existing host profiles**

1. Click the **Reports Tab**.
2. Click the **PDF Report** button on the Dynamic Secondary Navigation bar.
   
   The PDF Reports page displays a list of Host Profiles reports that have been created. Each report displays the following properties:
   - **Report Type**: The current Host Profile
   - **Hosts**: The number of hosts in the report
Mapping your business applications

You can use one of the following methods to map your business application:

- Creating SIs from discovered processes
- Collaborative application mapping
- Creating SIs in a pattern
- Creating a static application model

Creating SIs from discovered processes

Modeling a piece of running software as a Software Instance (SI) is a fundamental part of understanding and mapping your business applications. SIs to represent off the shelf software are usually created using base patterns shipped with BMC Atrium Discovery, or subsequent TKU releases. For custom or in-house developed software, you can either write TPL patterns and upload to the appliance using the Knowledge UI, or you can create them as part of the wider Collaborative Application Mapping (CAM) process.

If you do not want to write TPL, or you do not want to complete all the steps required to map a business application when you are only interested in creating a particular SI, you can use the Generate Software Instance Pattern page to generate a pattern to create SIs from discovered processes or services. Once the pattern has been generated and successfully activated, a new Software Instance is generated every time a process that matches the rules is discovered.

You need sufficient permissions to activate and deactivate patterns to create SIs from discovered processes, for example, the system user. If you have insufficient permissions, the Generate Software Instance Pattern option is not displayed in the Actions drop down.

This topic contains the following sections:

- To create an SI pattern by supplying conditions
  - To create an SI pattern from known processes or services
- To view generated SI patterns
- To modify generated SI patterns
- To delete generated SI patterns
To create an SI pattern by supplying conditions

1. Click the Applications tab in the Primary Navigation Bar.
2. Select either Create Software Instance pattern from Discovered Processes or Create Software Instance pattern from Discovered Services from the Application Mapping channel. The following page is displayed.

3. In the Model Discovered Process as Software Instance section, specify the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The type of software being modeled. The value entered in this field becomes the name of the pattern, and is prefixed with software_instance_model for the pattern module name. It is also the name of any SI created by the pattern.</td>
</tr>
<tr>
<td>Publisher</td>
<td>Publisher(s) of the SI.</td>
</tr>
<tr>
<td>Category</td>
<td>From the drop-down list, select the Category of the SI. The generated pattern is placed in the pattern module hierarchy according to this selection.</td>
</tr>
</tbody>
</table>

The fields are the same for discovered services.

4. In the Query Constructor tab, select the attributes, their values and a simple positive condition that identifies the processes of the SI and filters them out from other processes. The Query Constructor tab in this page is simpler than the usual Query Builder. Traversals are not supported and there is no refine search option for SI modeling.
5. Check the trigger condition in the Generate pattern section to ensure that the pattern that is to be generated is valid and that you have included all of the necessary conditions.

6. Ensure the results preview display only the processes you need for the SI.

7. Click **Generate TPL** in the Generate pattern section.
   The system generates the **pattern** for creating SIs of this type, loads it to the Knowledge Management module, activates it, and links the generated pattern to the model definition. From now on, a new Software Instance is generated every time a process that matches the rules is discovered.

To create an SI pattern from known processes or services
If you know that a particular process or services are running on a host, the list of processes or services is available from the Discovery Acess page for that host. From the list, select the processes or services and then select **Generate Software Instance Pattern** from the Actions drop-down menu. The **Generate Software Instance Pattern** page is displayed, populated with the conditions which match the process or service.

To view generated SI patterns

1. Click the **Discovery** tab in the Primary Navigation Bar and then click **Knowledge Management** in the top menu.
2. In the Pattern Modules section, filter the patterns to find the one you just generated:
   a. Select **only active** in the display option.
   b. In the Filter box, enter the SI type that you specified when created the pattern.
   c. Ensure that the **Generated** option is selected.
d. Click **Apply**.

e. If necessary, expand the pattern module groups by clicking the + sign.

f. Expand the pattern details by clicking the down arrow to the left of the pattern. The following example shows an example filtered pattern module.

```
2. 3. Click the pattern module name in the most left column to see the details of the pattern as in the following example.

To modify generated SI patterns

- At the filtered pattern module, as shown here:

Open the **Generate Software Instance Pattern** page. To do this, click the pattern module name link, labelled Software Mapping Definition.
• Modify the pattern details and triggering conditions and click **Generate TPL** in the Generate pattern section. The updated pattern is uploaded, activated and linked to the module definition.

**Changing pattern name creates new pattern**

The pattern name, which is entered in the **Type** field (it also determines the software instance type), acts as an ID for the pattern. If you change the pattern name, this does not update the name of the existing pattern, it generates a new pattern with the new name. As patterns do not deactivate other patterns, you must deactivate the previous pattern **manually**.

**To delete generated SI patterns**

1. View the pattern, that creates a Software Instance from discovered process.
2. In the **Actions** list, select:
   a. **Deactivate module** to deactivate the pattern and retain it on the system for potential further use.
   b. **Delete module** to entirely remove the pattern from the system.

**Collaborative application mapping**

The BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) application mapping guide is intended for anyone who wants to use BMC Atrium Discovery to map applications in their estate. This guide describes the application mapping methodology, how to gather and organize data and collaborate on a mapping prototype, and how BMC Atrium Discovery generates and maintains the resultant application map.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Related information</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn the concepts of Collaborative Application Mapping (CAM).</td>
<td>Introduction to collaborative application mapping</td>
<td>Understand the workflow of CAM and the roles that support personnel play in the process.</td>
</tr>
<tr>
<td>Find out some hosts and technologies involved in the application.</td>
<td>Gathering seed data</td>
<td>With very little input from the application owner, be able to begin searching and investigating the BMC Atrium Discovery datastore for clues about the application.</td>
</tr>
<tr>
<td>Use the seed data as a starting point to search the datastore for the software and hardware components that support the application.</td>
<td>Searching and investigating the datastore</td>
<td>Find your application without additional help from the application owner.</td>
</tr>
<tr>
<td>Build a static prototype application map.</td>
<td>Creating the prototype application map</td>
<td>Arrange the components found earlier into a manual application map to share with the application owner.</td>
</tr>
</tbody>
</table>
**Introduction to Collaborative Application Mapping**

**What is Collaborative Application Mapping?**

Knowing what business services are supported by which part of the IT infrastructure is essential to effective Business Service Management (BSM). Typically, most organizations have a list of the most critical applications, most of them tied to Service Level Agreements (SLAs). The goal of Collaborative Application Mapping (CAM) is to find out what hardware and software support which business applications, and to build the applications into service maps that can automatically be maintained. This is true for both enterprise and mainframe environments.

A dynamic, automatically maintained, effective application map enables you to understand the key relationships between how your business operates and the infrastructure that supports it. It also becomes the initial, crucial part of Service Impact Analysis by maintaining accurate service models for BSM.

This diagram illustrates how collaborative application mapping helps you build automatic service maps.

The CAM approach is designed to capture the rules that define where an application is running, not to simply define that information statically. This means that as you deploy the applications more widely in your estate, or you migrate them around your infrastructure, your service maps will stay current.

**Collaborative Application Mapping workflow**

The following is an illustration of the CAM workflow:
This diagram illustrates the collaborative application mapping workflow from inception to model.

**Video demonstrations**

Each stage in the CAM workflow contains a video demonstration to illustrate the process. The following table describes the number of each video and the corresponding goals for each stage described.

<table>
<thead>
<tr>
<th>Demonstration</th>
<th>Stages Described</th>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Video 1</td>
<td></td>
<td>Introduce the goal and characteristics of CAM.</td>
</tr>
<tr>
<td>Demonstration</td>
<td>Stages Described</td>
<td>Goals</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>-------</td>
</tr>
<tr>
<td>Video 1</td>
<td>This illustration represents Video 1 of the collaborative application mapping process. It is the same caption you see on the Applications tab.</td>
<td>Overview</td>
</tr>
</tbody>
</table>

**Overview**

| Video 2       | This illustration represents Video 2 of the CAM process. It is the same caption you see on the Applications tab. | 1. Obtain enough basic information from the application owner to begin searching and investigating the BMC Atrium Discovery datastore.  
2. Use the seed data as a starting point to search the datastore for the software and hardware components that support the application.  
3. Using the components found by searching and investigating, build a prototype application map. This is a manually built, static map that is used to assist in understanding the application. It is not the end goal of the process. |

**Seed Data; Search and Investigate; Prototype**

<p>| Video 3       | Share the prototype application map with the application owner by generating a preview report in PDF. This stage might involve revisiting the Prototype stage to redefine your working map based on feedback from the report. |       |</p>
<table>
<thead>
<tr>
<th>Demonstration</th>
<th>Stages Described</th>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>This illustration represents Video 3 of the CAM process. It is the same caption you see on the Applications tab.</td>
<td>Share</td>
<td>Configure rules by creating functional components that enable BMC Atrium Discovery to build and maintain the application map dynamically.</td>
</tr>
<tr>
<td>Video 4</td>
<td>This illustration represents Video 4 of the collaborative application mapping process. It is the same caption you see on the Applications tab.</td>
<td>Map Application (Part 1)</td>
</tr>
</tbody>
</table>
### Business roles

The following people are involved in the CAM process:

#### Application owner

The application owner is usually part of the application support team, and he might not have any knowledge of BMC Atrium Discovery. He handles trouble tickets and maintains the running application. Every application has an application owner; however, he takes direction from business owner of the application. The application owner has no stake in the mapping initiative, so getting
full cooperation can be difficult. He is too busy maintaining the application and can only spare small increments of time to collaborate on the maps. Consequently, the application owners are the greatest single cause of failure in the application mapping process. The goal is to create the application map without a large initial investment by the application owner. An effective application mapping process minimizes the reliance on the application owner, and any interaction must be as non-intrusive as possible.

In the business examples used in this guide, the application owner is named George.

Application mapper

The application mapper is part of the team responsible for BMC Atrium Discovery rollout and maintenance. He knows BMC Atrium Discovery well, particularly how to report, search, analyze, and use the application mapping user tools. The application mapper is responsible for executing and driving the mapping process, and to do this he must be familiar with the way in which business applications are put together, such as the roles of middleware, databases, web infrastructure, and message brokers in application architectures.

The application mapper's collaboration with the application owner should be as limited as possible, requiring only the basic information at the outset of the process and some feedback during the Prototype and Share stages of the process.

In the business examples used in this guide, the application mapper is named Mike.

Example

George, the application owner, administers the Friends application, a web-based corporate social networking application. Mike, the application mapper, is the BMC Atrium Discovery guy.

George is always busy responding to requests from the business owners, reacting to incidents, performing software updates, rolling out new versions of Friends, and so on. Friends is not the only application that George maintains.

Mike knows that George is the guy to speak to, but George does not need a map of the Friends application, because he has one in his memory. It is difficult for Mike to get George to commit much of his time to the application mapping initiative.

Mike must keep George on his side, because he is going to need George's cooperation in the future.

Demonstration
This illustration represents Video 1 of the CAM process. It is the same caption you see on the Applications tab.

Video 1 that follows provides an overview of collaborative application mapping, and highlights its approach and benefits.

Where to go from here
You start the process by identifying the application owner and gathering seed data.

Gathering seed data

The first stage in mapping the application is to gather seed data, a small sample of host names that are involved in the application or component names. The goal of gathering seed data is to provide just a few pieces of information to the application owner (typically communicated through e-mail or instant message) that are clues to help determine what to start investigating. The application mapper should not invest any significant effort upfront to share this knowledge with the application owner.

To gather seed data
Gathering seed data is the first of just a few interactions with the application owner. It sets the tone of the communication, so the process needs to be pleasant and not a burden. Some tips:

- Do not use a big questionnaire that would take the application owner a long time to complete.
- Do not book a big fact-finding meeting.
- Consider sending just an e-mail asking for a quick introduction to the product.
The information should be just detailed enough to understand what infrastructure supports which applications. For example, the application owner might specify that the application has two tiers, runs several hosts and uses a database. This is enough information for the mapper to get going. Later, in the mapping stage, this information will be coded in patterns and BMC Atrium Discovery will be able to keep the service models up to date.

**Example**

Mike determines that George, the application owner maintains an application named *Friends*, a web-based corporate social networking application, which is the next on his list of business applications to map.

He sends George the following e-mail:

```
George,

I see from my Applications and Owners report that you're the Application Support Lead for the "Friends" application. This is the application that we're mapping next for our BSM rollout initiative, which means that I'm going to be mapping this application in our BMC Atrium Discovery tool.

I don't need much from you - I know you're busy. Can you drop me a quick email with whatever information you think will be relevant? Just whatever you know right now. For example:

1. Do you know any hosts that it runs on?
2. What sort of app is it? A native executable? A J2EE web application? A multi-tier application distributed across several bits of middleware? Do you know any of the module or executable names?
3. Does it have a database? Do you know anything about that database off hand (Oracle/SQL Server, Name)?

I'll do some investigating in our BMC Atrium Discovery tool based on your reply and check back with you once I have something you can look at.

Thanks for your help,
Mike
```

George replies:

```
Mike,

Thanks for the mail.

Just offhand, *Friends* is a tomcat application.
```
The web module is named “friends.war”, and it's connected to a MySQL database. The test /UAT environment is running on lon416; I don't know right now where the PROD one is (our production maintenance guys handle the deployment there).

George

This is an example of effective seed data. It includes a host name, a module name, and a clue about what database to look for. This is more than enough to get started.

Demonstration

Video 2 demonstrates how to search and investigate seed data and start building the prototype map.

Where to go from here

After you have some seed data, you can begin to search the datastore for the components of the application.

Searching and investigating the datastore

The Search and Investigate stage of the mapping process is when you use the seed data obtained from the application owner as a starting point to figure out the structure of the application. You perform searches in the datastore based on search terms from the seed data, and use the search results as clues to do more searches until you eventually have the whole structure of the application. As you go along, you add what you find to the prototype application map. See the following example for a walk-through:

Before you begin

This stage requires certain skills and knowledge on the part of the application mapper. For more information, see the description of the application mapper role.
The premise of this stage in the workflow is that you find the infrastructure that supports the application by searching in the BMC Atrium Discovery datastore. This works well because, in general, an application mapping initiative starts when an BMC Atrium Discovery rollout is already far advanced. It is likely that the items supporting the application have already been discovered, and they just need to be found in BMC Atrium Discovery and brought together in an application map.

**Searching for application components**

You use what you have learned from the seed data provided by the application owner to begin your investigation. For example, you might find in your research that the Tomcat Software Instance (SI) depends on the database details. You might discover that a Tomcat search links it to certain modules. Other things to look for include:

1. Search for terms in your seed data.
2. Use what you find to try to understand the type of application it is, and add the items to the prototype, if appropriate.
3. Look in the items you have found for more search terms and search for them. For example:
   - Database names
   - Software names
   - Windows service names
   - Process names

Next, you gather all this information into subgroups to group application components into more refined blocks that represent the tiers in the application (for example, one for the web tier, and one for the DB tier). This is the beginning of your prototype.

**Example**

George has responded to Mike's e-mail. He tells Mike the following about the application:

- The name of the application is *Friends*.
- There is a Web component and a database, but he can't provide any further details.

Beginning with an index search, Mike starts to search the application for all the relevant infrastructure items, and continues his investigation until he can see how the application holds together, or until he needs to ask George more questions.
Demonstration

Video 2 demonstrates how to find out how to search and investigate seed data and start building the prototype map.

Where to go from here

As you build the prototype to find relevant infrastructure items, you must structure it to enable you to place components into groups that match the structure of the application. This refinement will enable you to get back to the application owner to check that the basic structure and content is correct.

Creating the prototype application map

While the application mapper searches in the datastore and investigates the results, he strives to build a picture of the application out of the IT components he finds. This picture is called the prototype application map, and is the goal of the Prototype stage in the Collaborative Application Mapping (CAM) process.

A successful prototype is crucial to CAM, and it fulfills two purposes:

1. Provides a logical framework for the application mapper to capture what he finds about the application, and is a vital part of his toolset.
2. Becomes something that can be shared with the application owner and to solicit feedback, thereby creating a very effective communication tool. See Sharing the prototype application map for more details.
Before you begin

Before you can build the prototype map, ensure that you have started searching and investigating the items in the datastore that should be added to the prototype, using the techniques described in Searching and investigating the datastore.

To create the application prototype

The prototype is built using the Manual grouping functionality in BMC Atrium Discovery. The application mapper creates a manual group for the application he is mapping. The group is the workspace for the application map.

The prototype is built in the group by creating subgroups to correspond to the application tiers, also known as functional components. A functional component is a part of an application that you want to map as a separate section. Generally, an application has one functional component for each tier (web, business logic, and database), but there are some more complicated applications that might need multiple functional components per tier. For example, if an application consists of a web layer, some message-driven Enterprise JavaBeans (EJBs) in one set of J2EE containers, a message broker, and a spring-based Plain Old Java Objects (POJO) service in a different set of J2EE containers, you would consider creating four functional components to represent the application.

As the mapper finds items in the datastore that he wants to add to the prototype, he adds them to the appropriate subgroup that corresponds to the functional component that he wants the item to be modeled in.

Notes are a key component in the CAM process, even at this early prototyping stage. Writing down observations and questions in this stage helps you maintain your understanding of the application as the prototype potentially becomes increasingly complicated. The questions are useful for soliciting feedback from the application owner when the time comes.

The Application Mapping channel enables you to create a manual group to use in application mapping and provides a list of links to existing manual groups.

To create a manual group

1. In the Application Mapping channel, enter the name of the new group in the Create group for mapping field. The period (.) double quote (") and dollar ($) symbols cannot be used in a group name.
2. Click Create.
   The Home Page for the new empty group is displayed.
To add a subgroup to a group

1. Select the nodes to add to the new subgroup. You can use the select tools to help with this: (Select: All | None | Invert).
2. Click the Actions button to display the Actions menu, and select the Manual Groups item.
3. If the required parent group already exists, click the entry in the displayed list to enter its name in the Group field (or enter the name of the group manually). After the name is entered, the non-matching groups are removed from the displayed list and the Subgroup entry field is enabled.
4. If the parent group does not already exist, create it by entering a name in the Group field.
5. Enter the name of the subgroup into the Subgroup entry field.

To view a group page

Click the name of the group in the group list. The Home page for the group displays the Application Mapping tab.
For more information, see Viewing created groups.

Example

In the following example, Mike continues where he left off in the previous stage, having gathered enough information from his searching the Discovery datastore for seed data given to him by George.

After locating hosts, software instances (SIs), and DBs, Mike gathers all this information into subgroups to group application components into more refined blocks that represent the tiers in the application (one for the web tier, and one for the DB tier).

1. Create a workgroup for the application *Friends*, the application name that George provided as seed data.

   ![Screen Illustrates How Application Mappers Create a Workgroup for the Friends Application](image)

   This screen illustrates how application mappers create a workgroup for the Friends application.

2. Perform an index search of the BMC Atrium Discovery datastore for *Friends*.

   ![Screen Illustrates the Results of an Index Search for Friends](image)

   This screen illustrates the results of an index search for *Friends*.
   The results of the search indicate that two SIs and two software components were located. Mike clicks through to the SI details, and notices two Tomcat application servers, likely representing the Web component of the Friends application that George mentioned previously.

   ![Screen Illustrates the SI Details of the Friends Application Search](image)

   This screen illustrates the SI details of the Friends application search.
Drilling into each of the servers shows:

- Software components, which are modules that run inside middleware (for example, J2EE EARs and WARs, .Net applications, IIS virtual directories, and so forth).
- Client to server communications to a MySQL database
- A dependency to that database (SQL Database Schema mellon for one server, one appended with _dev for the other)

3. From the **Actions** menu, select **Manual Groups > Friends**.
4. In the Subgroup field, type **web** and click **Create**.

![Screen illustrating how to add the software component to a subgroup of the Friends application.](image)

This screen illustrates how to add the software component to a subgroup of the Friends application.

5. Perform an index search of the BMC Atrium Discovery datastore for Mellon (the name associated with the database).
6. Click on the **Database Detail** link.
7. From the Database Detail list, select all instances of the database and create a manual subgroup in the Friends group named **db**.

![Screen illustrating how to add the database instances to a subgroup of the Friends application.](image)

This screen illustrates how to add the database instances to a subgroup of the Friends application.

8. View the manual group to assess the details of the current prototype.

![Screen illustrating the created group with the defined subgroups representing the prototyped application.](image)

This screen illustrates the created group with the defined subgroups representing the prototyped application.

9. If necessary, add notes at the group, subgroup, or top level, or **drag and drop** items to rearrange them in the map to provide better context in the hierarchy.
Demonstration

This illustration represents Video 2 of the Collaborative Application Mapping process. It is the same caption you see on the Applications tab.

Video 2 that follows illustrates the first three stages in the CAM process and shows how, by starting with seed data, the application mapper can begin searching the discovery datastore and building the prototype map.

Where to go from here

After you have created your prototype, you should have enough context in your applications to begin sharing information with the application owner in a PDF report. Creating a preview with context-driven notes enables the application mapper to communicate with the application owner efficiently on the map building process. Then mapper can apply the owner's feedback to refine the map before building it our further.

Dragging and dropping items

You can drag and drop items between groups and subgroups. This includes both nodes and multi-kind nodes. When you drag and drop these items, the group is recreated and the nodes that are moved are copied into the group.

To drag and drop items

1. On the Group page, click the four-pointed icon selector corresponding to the node you want to move and drag it into the new group.
This screen illustrates how to drag and drop items from one subgroup to another.

Click the four-pointed icon

You must click the four-pointed icon to grab the item to be dragged. You will notice that the icon is moving with the cursor motion during the dragging process. You cannot try to click the selection itself.

2. To delete an item from a group, click the four-pointed icon selector corresponding to the node you want to delete and drag it to the trash can icon.

This screen illustrates how to drag and drop a node to the trash can to delete it.

If you have removed all items from a subgroup, a message displays in the group that the subgroup is empty.

This screen illustrates an empty subgroup where all items have been either dragged and dropped to another subgroup or to the trash can to be removed.

If the subgroup is empty, the trash can icon no longer displays in the subgroup.
Sharing the prototype application map

The **Share** stage involves using a dynamic PDF report to make collaboration more effective and informative. The report, which contains a preview of the mapping prototype, enables you to share the captured data you have collected in groups and subgroups and solicit feedback to determine any missing pieces. The **application mapper** provides the report to the **application owner** so that the owner can review the contents of the discovered applications and determine whether the prototype map should be refined further.

**Recommendation**

Although you can create a PDF preview of the application map at any time, typically you perform the sharing exercise at least twice; once after the initial prototype, and once after the application mapper has added **rules** to the prototype and has mapped the application. This enables you to gather feedback before you evaluate your initial data gathering and finalize the map.

**Before you begin**

Ensure that you have captured enough information to effectively communicate any progress or questions you have with the current prototype. Notes can be added for groups, subgroups, or at the top level to capture any questions or comments you want to share with the application owner in the preview PDF. You also have the option to set the level of detail in the report (the default is **Medium**), should you want to include more or less information, depending on how much input you want from the owner. See the following example for more information.

**To add notes to a group page**

The top of the page includes a Notes section where you can add and edit notes on the group. The notes can be typed in simple wiki markup, which enables you to display the text in HTML format. You can include paragraph formatting, hypertext links, and create simple or complex lists to the level that you require.

1. Click **Edit** to the right of the Notes section and type or copy the text in the **Notes** field that you want to define for the group.

You can create new paragraphs, or type in-line mark-up to display wiki text in various formats. You can click the **Show formatting help** link to display a page that describes the syntax to use for all types of inline formatting. For example, the following screen illustrates a complex list:
1. This screen illustrates how to create a list format in the group notes.

2. Click **Save**. The notes are displayed in the format that you specified. The following screen illustrates an example:

![Example Screen]

This screen illustrates how to include notes in various text formats on the Group page.

3. If necessary, click **Edit** again to go back and revise the text.

**To create a preview report**

1. On the Group page, click the **Contents** tab.

2. Select the level of detail you want in the report using one of the following methods:

   a. Click an area on the Reporting Level bar, where the lowest level is located on the left and the highest level is located on the right (the default setting is **Medium**), as illustrated in the following screen:
This screen illustrates how to adjust the reporting level of the group PDF report.

b. From the Actions menu, choose one of the following reporting levels:

<table>
<thead>
<tr>
<th>Level</th>
<th>Type of report generated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set</td>
<td>Details the names of the nodes in each group and a list of attributes for each node, and displays functional components with all relationships to other components.</td>
</tr>
<tr>
<td>PDF</td>
<td></td>
</tr>
<tr>
<td>Detail</td>
<td></td>
</tr>
<tr>
<td>High</td>
<td></td>
</tr>
<tr>
<td>Set</td>
<td>Details the names of the nodes in each group and a list of important attributes for each node. This is the default reporting level.</td>
</tr>
<tr>
<td>PDF</td>
<td></td>
</tr>
<tr>
<td>Detail</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td>Set</td>
<td>Details the names of the nodes in each group, but no further information on the nodes. You might choose this level, for example, if you have complex diagrams in your report where the links or relationships are not readable. This would display only the functional components and not their relationships to other components.</td>
</tr>
<tr>
<td>PDF</td>
<td></td>
</tr>
<tr>
<td>Detail</td>
<td></td>
</tr>
<tr>
<td>Low</td>
<td></td>
</tr>
</tbody>
</table>

An example of all three levels combined in one report to represent a three-tier application is shown here.

**Note**

Reports do not show functional component section headings for functional components that have no messages to display.

3. From the Actions menu, select Generate PDF Report.

   A confirmation displays indicating what types of content will be included in the report.

   **This screen illustrates a notification of what content will be included in the generated PDF.**

4. From the status link at the top of the Group page, click the Reports Tab link.
5. From the **Reports** tab, click **PDF Reports**.
6. Select the Group Report in the list and click **Download**.

**Example**

After Mike has organized groups and created a prototype, he wants to collaborate on his findings with George to ensure that the right information is being accounted for. As an important first step in this process, he uses both the drag-and-drop and notes features to tailor the PDF preview report in the way that might make the collaboration more effective.

1. In the **Notes** field, type a summary of the prototype in wiki markup and questions you want to ask the application owner before finishing the prototype.

![Notes example](image)

**This picture illustrates a set of notes to the application owner that will be included in the PDF.**

2. Select a level of detail for the PDF report.

![Recommendation](image)

**Recommendation**

Choose a low level of detail if there are complex diagrams in the report where the links or relationships are not readable. This displays only the functional components and not their relationships to other components.

3. Generate the report.
3. This picture illustrates how to generate the PDF.

4. Review the report. The following example shows the first page of the report with the Bookmarks pane open and the notes that were applied previously displayed at the top:

This picture illustrates the first page of the report with the Bookmarks pane open and the notes that were applied previously displayed at the top.

5. Attach the PDF to an e-mail and send it to the application owner.

   Based on the feedback from the George, Mike can go back and refine his prototype. He can perform a PDF preview as many times as he wants until he and George are satisfied with the prototype.

Demonstration
This illustration represents Video 3 of the collaborative application mapping process. It is the same caption you see on the Applications tab.

Video 3 that follows demonstrates how the application mapper begins his collaboration with the application owner by creating and sharing the prototype map.

Where to go from here
After you are satisfied with the feedback you have received from the PDF preview, you are ready to start mapping the application.

Mapping the application

After you have generated a prototype and have solicited feedback through a PDF preview, you are ready to move to the next stage and map your application. In your prototype, the specifications are only in your internal configuration. You have to ensure that they are created into patterns so that, during discovery, BMC Atrium Discovery can build the Business Application Instances (BAIs) and push them into BMC Atrium Configuration Management Database (BMC Atrium CMDB). The Map Application stage transforms your static map to a dynamic mapping structure that enables BMC Atrium Discovery to recognize and maintain the application during and after your scans. The stage is divided into two steps:

2. Identify the environment by dividing into instances through named values.

Demonstration
See the following videos to learn how to map the application:
Video 4 demonstrates how to configure rules by creating functional components that enable BMC Atrium Discovery to build and maintain the application map dynamically.

Video 5 illustrates how to divide the application into instances to enable BMC Atrium Discovery to identify the environment to build and maintain the application map dynamically.

Video 6 demonstrates how to generate patterns to create the model in BMC Atrium Discovery.

Where to go from here
The first step in mapping the application is to create rules that help BMC Discovery understand how to build and maintain the application. For example, if you find a database named Payroll, then that database is supporting the Payroll business application. That is the rule that BMC Atrium Discovery uses to maintain the application dynamically. To do this, you create functional components that combine similar functionality into logical chunks or blocks of information. Based on the traversal patterns you used to create groups and subgroups, functional components give more context for the map (for example, defining the business logic layer), thereby enabling BMC Atrium Discovery to dynamically discover and maintain your application.

Creating functional components

To create a functional component, you define each tier in your application with a set of rules that BMC Atrium Discovery can use to find the contents of those components and then ultimately find the entire application. These rules are known as Functional Component Definitions. BMC Atrium Discovery can infer from the Functional Component Definition the necessary details to create the model. In this manner, BMC Atrium Discovery enables you to model simple to sophisticated applications without the need to write The Pattern Language (TPL). There is no need to download and then upload a TPL file. BMC Atrium Discovery automatically activates the pattern and submits it into knowledge management.

Before you begin

The end purpose of the Functional Component Definitions is to generate TPL patterns from your definitions. Patterns always have a trigger; in Collaborative Application Mapping (CAM), the trigger is supplied by a Functional Component Definition. You can only trigger on a positive occurrence, so one of the Functional Components in your map must have a positive operator (for example, contains, is or starts with, but not does not contain or is greater than). If no positive operator is set, the following error message is displayed:

At least one top level Functional Component Definition must contain a simple, positive condition.

You cannot create conditions based on date or time attributes.

⚠️ Non-ASCII Unicode characters in CAM

In Collaborative Application Mapping (CAM), if you create components such as group names or functional components, that contain non-ASCII Unicode characters, the Business Application Instance (BAI) that results from running the pattern displays with unreadable characters.

To add a Functional Component Definition

1. On the Application Mapping tab, click Add New Functional Component Definition for each tier in your application.
2. In the **Add Functional Component Definition** box, type a name and any notes that apply to the component and click **Apply**.

3. The page is refreshed to show the new functional component definition. Click the name to edit the functional component definition.
   
   The Functional Component Definition opens by default on the **Settings** tab.

4. Choose one of the following options from the **Start with** menu to define how to find the contents of the functional component:
   - Software Instance
   - Database Detail
   - Detail
   - Software Component
   - Application Instance

5. Choose one of the following options from the **Type** menu to define the functional component:
   - **Required** — the functional component *must* be present for the BAI to be created. If you delete this functional component, the BAI is deleted too.
   - **Optional** — the functional component may exist but is not required for the BAI to exist. The functional component is only created if the appropriate information is found.
   - **Always Present** — similar to Optional though the functional component is always created even if the associated content is not found. For example, you know there must be a database but have failed to discover it, possibly because you do not have rights or cannot scan the endpoint. This type of functional component exists so it is possible to build a complete functional model. It is not removed until the BAI is removed.
   - **Working Step Only** — enables you to navigate through the model without creating a functional component. You can think of this as a placeholder for named values to be used for modeling other functional components.

6. In the Query Builder section, build and refine the queries to better define how BMC Atrium Discovery can find the functional components.

   ![Note]
   
   The error message "Same node in more than one functional component definition" is displayed if your query returns nodes that are already returned by another functional component query.

7. View the results in the Results section at the bottom of the page and go back and refine your queries as necessary.
   
   You can also define traversals, helping you browse around your data. For example, you could traverse from a set of hosts to the software running on all of those hosts.
Recommendation

Do not automatically choose the first entry in the query list. Sometimes you need to scroll down the list to get the best choice.

8. Click **Refresh results**.

**Refresh results**

It is important to refresh the results to ensure that BMC Atrium Discovery will pick up any changes in the datastore resulting from changing the contents of groups.

**SI to SI traversals**

The following relationships are among those available when specifying a traversal from Software Instance (SI) to SI:

- SIs observed to be communicating with this SI
- Server SIs that this SI is communicating with (Client to Server Comms)
- Client SIs that are communicating with this SI (Server to Client Comms)
- Peer SIs that are communicating with this SI (Peer to Peer Comms)

The last three are explicit, meaning they are built by BMC Atrium Discovery for later use. Examples of explicit relationships are those created by patterns that parse application configuration files to find communicating hosts.

The first relationship is a pseudo-traversal based on the communicating SIs function. While the pseudo-traversal might be useful where relationships corresponding to the explicit relationships do not exist, it should be used with caution due to its potential impact on performance. These relationships can be created from the output of commands such as `netstat`.

The first relationship is available only in the the **Traverse to** dialog, and not the **Filter** dialog. You cannot filter by the first relationship because it is not a key expression.

**Undoing actions**

Creating Functional Component Definitions involves making anything from simple, discrete type changes to much more complex traversals and conditions, and sometimes during the process you might lose track of what you have done. Other times, you might want to undo a step to change your orientation and start again. The Undo feature enables you to cancel the one or a series of recent actions, and it displays the result in a banner at the top of the Functional Component Definition page.

To undo an action, click the **Undo last change** link to the right of the Functional Component Definition page. You must perform an action on the page before the **Undo last change** link displays.
In the following example screen, the **Type** field has been changed to **Required** and the link has displayed on the page:

![Example Screen](image1)

Hovering over the Undo link displays a tooltip that displays the action that will be undone. After the link is clicked, an information banner displays at the top of the page indicating the change.

![Example Screen](image2)

The Undo action applies to a single state change or the link can be clicked multiple times to cycle back through multiple changes. The Undo sequence persists until you log off the system; therefore, you can navigate away from page as necessary. The Undo action is very definitive, so that even if you made a single character change in a query, for example, a single Undo action would only retrieve that character (making it appear that nothing has actually changed). The action works for all functions on the Functional Component Definition page, including traversals and deletions.

**Example**

Functional components help BMC Atrium Discovery find your applications by breaking them down into more defined blocks of information. From there, you define those components into rules that BMC Atrium Discovery can use to find the components and the application that uses them. You also used named values to differentiate and identify each component.
Functional components can form hierarchical structures (for example, database, application server, and client tiers), but can also be used to form parallel blocks of data flows. The structure does not have to be tiered. They can contain software instances, business applications, and software components. For example, in the two-tier Friends application structure, the Web tier is a functional component, because it can help express redundancy to measure the impact on data. The database tier is also a functional component.

Mike, the application mapper, has completed the first round of information gathering and has collaborated with George, the application owner, on the prototype map using a PDF preview report. Mike's next step is to create functional components.

1. Create a functional component for each subgroup or tier (Web and DB) that you created in the prototype. Click Web.

   ![Functional Component Definition: Web](image)

   Next, you begin to define the rules that BMC Atrium Discovery will use to identify the applications.

2. Select the type of data to start with.

   ![Functional Component Definition: Web](image)

3. Select the type associated with the data.

   In this example, Required is selected for the Web tier because it represents a crucial Web application that must be present in the map.

   ![Functional Component Definition: Web](image)
4. Run queries with conditions to tailor the Results list to show only Software Components with the name Friends.

![Image of query results]

5. Click Refresh results.
The two software components that include the name friends are displayed in the Results pane.

**Recommendation**

Remember that you can click the **Undo last change** link to revert back to an earlier action if you make a mistake at any point in the procedure.

6. Repeat steps 1 through 5 for the database component, while adhering to the following conditions for the database:
   
   - Remember that the database has a relationship, so start with Web (Software Component) and select **Required** for the **Type** field.
   - Traverse from Software Component to Software Instance through the **Software Instance this Software Component is running on** relationship.
   - Traverse from Software Instance to Database Detail through the **Detail depended upon by this Software Instance** relationship.
Demonstration

Video 4 that follows demonstrates how the application mapper can begin building rules in his prototype map by adding functional components, the first step in mapping the application.

Where to go from here

The second step in the mapping stage is to divide the application into instances. In order for BMC Atrium Discovery to link components together into a single instance of the application, you must derive an identity that is the same in each functional component. Named values provide unique characteristics to differentiate each instance within the application.

An example of the use of named values is to determine the identity of your application (for example, Development and Production) and provide meaningful abbreviations for a Proof of Concept. You can manipulate the values to convert a process user name like slcPROD and slcDEV to become Prod and Dev.

Using and manipulating named values

Named values enable you to provide the identity in human-readable naming conventions that help you differentiate between instances. However, no type conversions can be performed in the named values user interface, so you can only compare values of the same type. When you select a named value to use, only those that can be compared are displayed in the drop-down lists. Having fewer named values displayed than you are aware of might initially appear counter-intuitive, but it prevents comparing unlike types. Similarly, string operations are only permitted on strings.

Where such a restriction is enforced in the user interface, a green question icon is displayed, which links to the following explanations for the corresponding restriction:

- **Why can’t I apply transformations?** Transformations are manipulations of string attributes, so they cannot be applied to non-string attributes.
• **Why can't I see all named values here?** Named values can be compared only with named values of the same type (for example, string with string, number with number, and so forth). Therefore, if you choose a named value of one type, not all other named values might be available for comparison.

• **Why can't I select named values here?** Named values can be compared only with named values of the same type. If there are no other named values of that type, then no values can be selected. Also, named values will not display from other trees in the drop-down list, or for named values defined on the Functional Component Definition that you are adding a condition for.

You cannot create named values based on date or time attributes.

The following table describes the transformations that you can apply to named values.

<table>
<thead>
<tr>
<th>Transformation type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>append string [constant]</td>
<td>Append the specified string to the named value string.</td>
</tr>
<tr>
<td>left [index]</td>
<td>Use the first ( n ) characters from the named value string, where ( n ) is specified as the index.</td>
</tr>
<tr>
<td>lower</td>
<td>Convert any uppercase characters in the named value string to lowercase characters.</td>
</tr>
<tr>
<td>middle [index1, index2]</td>
<td>Use a section of the named value string. The character to start from is specified with ( index ); and the length of the string is specified with ( length ).</td>
</tr>
<tr>
<td>prepend string [constant]</td>
<td>Prepend, or add the specified string to the start of the named value string.</td>
</tr>
<tr>
<td>regular expression [regex, default]</td>
<td>Use a regular expression to extract matching characters from the named value string. See the note below.</td>
</tr>
<tr>
<td>right [index]</td>
<td>Use the last ( n ) characters from the named value string, where ( n ) is specified as the index.</td>
</tr>
<tr>
<td>split on [index, string]</td>
<td>Extract the group of characters from the named value string using ( string ) as a delimiter. The index specifies which group to extract.</td>
</tr>
<tr>
<td>table [default, (key, value)...]</td>
<td>A table of key and value pairs. Where the named value string matches the left hand value, it is converted into the value on the right. The default value is used where the named value string does not match any of the values in the left hand column.</td>
</tr>
<tr>
<td>trim</td>
<td>Remove leading or trailing whitespace characters from the named value string.</td>
</tr>
<tr>
<td>upper</td>
<td>Convert any lowercase characters in the named value string to uppercase characters.</td>
</tr>
</tbody>
</table>

The following table shows examples of each of the transformations described in the previous table:
### Transformation type

<table>
<thead>
<tr>
<th>Transformation type</th>
<th>Example input</th>
<th>Transformation</th>
<th>After transformation</th>
</tr>
</thead>
<tbody>
<tr>
<td>append string</td>
<td>webdav</td>
<td>&quot; instance&quot;</td>
<td>webdav instance</td>
</tr>
<tr>
<td>[constant]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>left [index]</td>
<td>webdav</td>
<td>3</td>
<td>web</td>
</tr>
<tr>
<td>lower</td>
<td>WebDav</td>
<td>—</td>
<td>webdav</td>
</tr>
<tr>
<td>middle [index1,</td>
<td>WebDav inst</td>
<td>4, 9</td>
<td>Dav in</td>
</tr>
<tr>
<td>index2]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prepend string</td>
<td>WebDav</td>
<td>&quot;My &quot;</td>
<td>My WebDav</td>
</tr>
<tr>
<td>[constant]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>regular expression</td>
<td>WebDav</td>
<td>([A-Z]+)$, Unknown</td>
<td>Unknown</td>
</tr>
<tr>
<td>[regex, default]</td>
<td>wlc=PROD wlc=TEST wlc=UAT</td>
<td></td>
<td>PROD TEST UAT</td>
</tr>
<tr>
<td>See the note below.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>right [index]</td>
<td>webdav</td>
<td>3</td>
<td>dav</td>
</tr>
<tr>
<td>split on [index,</td>
<td>module on App Server on rhe15-tomcat55</td>
<td>1 &quot; on &quot;</td>
<td>App Server</td>
</tr>
<tr>
<td>string]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>table [default,</td>
<td>deb-tomcat55</td>
<td>Default None</td>
<td>None</td>
</tr>
<tr>
<td>(key, value)...</td>
<td>rhe15-tomcat55</td>
<td>rhe15-tomcat55</td>
<td>Dev</td>
</tr>
<tr>
<td></td>
<td>w2k3-tomcat55</td>
<td>w2k3-tomcat55</td>
<td>Prod</td>
</tr>
<tr>
<td>trim</td>
<td>&quot; webdav &quot;</td>
<td>—</td>
<td>webdav</td>
</tr>
<tr>
<td>upper</td>
<td>webdav</td>
<td>—</td>
<td>WEBDAV</td>
</tr>
</tbody>
</table>

⚠️ **Using regular expressions in transformations**

Using a complex regular expression (one that takes a long time to process) as a transformation can render the user interface unresponsive to all logged in users. For information on regular expressions in BMC Atrium Discovery, see [Writing efficient regular expressions](#).

To create a named value

1. On the **Application Mapping** tab, click **Defined Named Value** corresponding to a functional component.

2. In **Named Value Editor**, choose a value from the **Add New Transformation** menu. This creates a transformation that applies a definition to the named value.
2. This screen illustrates how to add a new transformation to a named value in the Named Value editor.

3. Click **Apply**. The transformation displays in the field below the **Add New Transformation** menu, marked with an **X**.

4. In the **Transformation** field, type a value to see how the transformation applies to the corresponding name. In the following example, typing MYNAME in the field displays myname in the field corresponding to the selected transformation, indicating that the lowercase transformation works as expected.

   If you type a value that exists in the datastore, the system will fetch the values and match the text, and then display those on the Editor. The following screen illustrates values for the process User Name being pulled from actual data in the datastore for each instance and applying the selected transformation to each one (in this example, a lowercase, prepended "I am the").

5. Click the **X** near the transformations you have validated and want to remove.

6. Repeat Steps 1 and 2 to add transformations that better define the value as necessary.

7. In the **Make available as** field, type the name of the identifier for an instance, select the **Is identity** check box, and click **Apply**.
This screen illustrates values for the process User Name being pulled from actual data in the datastore for each instance and applying the selected transformation to each one.

**Is identity option**

The **Is identity** option applies only to one value.

After you have created values with names, you can also manipulate the values so that BMC Atrium Discovery can:

- provide the identity in human-readable naming conventions that help you differentiate between instances
- link all the components of an application instance
- make the names more logical to identify the application

**Example**

In his questions to George in his PDF, Mike needs to know if there are both a Production and Development instance in the Friends application. Rather than wait for an answer and be blocked from further progress, he knows that there are two database instances (mellon, and mellon_dev), which clues him that the latter is the Development instance, and the former might be the Production instance.

This screen illustrates the Group that indicates a development instance for the database.
Based on this assumption, Mike begins to define a named value for the database tier in the application to differentiate across the Production and Development instances. He notices that, in the Results section, there are some names assigned to the instances that would require stripping the names to match the Friends application. So, because the database has instances, he decides to check that attribute first.

1. In **Named Value Editor**, click **show all** in the Attributes section to expand the list of attributes.

2. Click the **Instance attribute**.

   The Sample values show the mellon names that he was looking for.

   ![Image of Named Value Editor](image1.png)

   **This screen illustrates the Group that indicates a development instance for the database.**

3. Transform the mellon instance to the Production instance, and mellon_dev to the Development instance.

   ![Image of Named Value Editor](image2.png)

   **This screen illustrates how to transform both names of the database instances.**

4. Make the value available as the identifier `friends_db_id`.

   Next, Mike wants to run a **PDF preview** and verify with George that the model is taking shape. The PDF displays one page per functional component and a tree view of the functional component (which can also be used as an index). You can control the level of
reporting in the PDF to respond to requests for more or less information, evaluate a bigger picture of the environment, or drill down to the details of the functional components and their relationships to all other nodes.

5. From the **Generate PDF Report** dialog, choose the visualization detail you want to display in the report by selecting one of the following option buttons:

- **No visualization** — generates a report with no illustrative examples of the functional components and their relationships to other hosts.
- **Functional Components only** — generates a report with descriptive and visual information on functional components, but not about their relationships with other hosts.
- **Functional Components and related hosts** (the default) — generates a report with descriptive and visual information on functional components, including relationships with other hosts. For an example visualization, see page 10 of the example report.

The **Generate PDF Report** dialog displays any guidance or warning messages associated with your application mapping exercise to this point.

This screen illustrates a notification of prior errors relating to the application map before printing to PDF.

6. If necessary, go back and make changes to the mapping in response to any messages.
7. Click **OK**.
8. Download the PDF and review it.

The PDF now contains visualizations of both the database and Web tiers of the Friends application. In the following example, the functional components are indicated by the gray boxes for both the database and Web tiers, and the rules that Mike defined for each tier are labeled inside the boxes:

This screen illustrates the visualizations included in the PDF preview for both the Web and database tiers.
Checking his e-mail, Mike notices that George has responded to his questions that he noted in his earlier PDF report, confirming that the Friends application contains both a Production and Development instance. This validates Mike's findings in the PDF report and gives him the confidence to send his latest report to George.

9. Attach the PDF to an e-mail it to George for his feedback.
George is impressed that Mike was able to generate such a rich report with all the details of the application. More importantly, George is happy that Mike was able to map the application accurately and efficiently with little help.

Demonstration

This illustration represents Video 5 of the collaborative application mapping process. It is the same caption you see on the Applications tab.

Video 5 that follows demonstrates how the application mapper can begin dividing the application into instances, the second step in mapping the application.

Where to go from here

After you defined the rules for your application through named values, the final step in the mapping stage is to generate a pattern from the application definition. This step turns the results into a form that BMC Atrium Discovery can use at during normal discovery when you scan your estate, and so it can build the functional components and Business Application Instances (BAIs) and push them into BMC Atrium Configuration Management Database (BMC Atrium CMDB).

Generating patterns for the application model

After you have created named values, you are ready to generate The Pattern Language (TPL) to encode all the details of your map into BMC Atrium Discovery. The pattern is automatically uploaded to BMC Atrium Discovery through the knowledge management UI, after which it will run during discovery and build the appropriate model whenever your application is discovered. BMC Atrium Discovery can discover and automatically build Business Application Instances (BAIs) based on the generated patterns that contain Functional Component Definitions, and it maintains the definitions in patterns so that it is ready to use in your environment.
Before you begin

Patterns are not generated for Software Instances (SIs), nor can you create custom SIs. To create custom SIs, you can use one of the template patterns available for download.

To generate patterns for the application model

1. From the **Application Mapping** tab on the Group page, click **Generate TPL**.
   BMC Atrium Discovery automatically activates the pattern that has been generated and submits it into the knowledge management UI, as illustrated in the following screen:

   ![Screen Illustrates Pattern Submission](image)

   **This screen illustrates the pattern automatically being submitted to Pattern Management.**

2. From the **Discovery** tab, click **Recent Runs**.
3. Click **Rescan Now** to scan the architecture again.
4. Click the **Applications** tab.
5. In the Application Summary section, click **Application Instances**.
   The applications are displayed in the list.

   ![List of Application Instances](image)

   **This screen illustrates the listed application instances.**

6. Click one of the application instances in the list.
   The Application Instance page displays the complete BAI, detailing the functional components and software instances in the application.

   ![Full BAI on Application Instance Page](image)

   **This screen illustrates the full BAI detailed on the Application Instance page.**
Example

In the second step of the mapping stage, Mike was able to identify the name of the Friends application (representing the transformed values) to divide the results into different instances (PROD and DEV). Now, at discovery time, BMC Atrium Discovery can use the transformed identity to split discovered results into different instances of the Friends application being mapped. Where Mike left off, the results of the identity change are displayed from the results of the Web and DB functional components that Mike has mapped.

This screen illustrates the results of the identity assigned to a named value for the Web and DB functional components and differentiate instances in the application.

The final step is for Mike to encode the definitions into TPL.

1. Click Generate TPL.
   The Knowledge Management page displays showing the changes being applied to BMC Atrium Discovery. When the pattern is finished generating, a banner displays at the top of the Knowledge Management page indicating that the requested changes were successful. The Friends Application Model is listed on the Knowledge Management page.

   This screen illustrates the results of the generated TPL and shows the Friends application has been modeled into a pattern.

   2. Scan the estate to take a look at the Friends application instances.

   3. Scan any Mainframe instances and ensure that you see the number of instances you expected, and remember that you can filter the list to get a more granular view.
When Mike performs a scan, he notices that both the Friends instances DEV and PROD have been built. Drilling down into the DEV instance, he can see the functional components, software instances, software components, and hosts that make up the application. He can also see that the new pattern (applicationmodel.Friends.build_Friends) has been generated.

This screen illustrates the results of the generated TPL and shows the Friends application has been modeled into a pattern.

Mike has successfully mapped the Friends application and created the model in BMC Atrium Discovery.

Demonstration

This illustration represents Video 6 of the collaborative application mapping process. It is the same caption you see on the Applications tab.

Video 6 that follows demonstrates how the application mapper can generate TPL to encode the application map into patterns and complete the model.
Where to go from here

After you have generated TPL, you have mapped your application and created the model that will be used and maintained in BMC Atrium Discovery. At this point, you might also want to create a working back-up of your application map. If so, export a single group or set of groups containing Functional Component Definitions you have created. The exporting function can also be a useful tool for troubleshooting any problems you had in the mapping process, because it gives you a full history of the definitions that you applied in the map.

Exporting and importing application mapping definitions

You can export a single group or set of groups containing Functional Component Definitions. You would do this typically to back up the functional definitions and notes in the application that can be restored to the same appliance. The back-up can also serve as a:

- working back-up during mapping
- application mapping back-up in case of required appliance maintenance (for example, a model reinitialization)

You can also export application mapping definitions and move them to another system.

The same application mapping definitions can be deployed in different environments, such as production, development, and test. If you want to edit a definition that exists in more than one environment, you must do one of the following:

- Edit the application mapping definition in each environment where it is present. This is the recommended procedure to edit definitions.
- Edit the application mapping definition in one environment and export or import the edited definition to other environments. This is not the recommended procedure because, before attempting to export or import the edited definition, you must destroy the existing definition on the destination environment. If BMC Atrium Configuration Management Database (BMC Atrium CMDB) synchronization is in the continuous mode on the destination environment, destroying the existing definition causes the BMC_Application configuration item (CI) to be marked as deleted in BMC Atrium CMDB. However, after a successful export or import, the BMC_Application CI is inserted into BMC Atrium CMDB again.

Before you begin

The exported mapping definition contains a subset of what is in the group page, it does not contain subgroups (even when importing to the same appliance). A global note is preserved with the names of all groups, as well as notes contained in the global group and any subgroups. Subgroups themselves and subgroup PDF-level settings are not included in the exported information. The nodes contained in the global group or subgroup are not exported.

You can export mapping definitions using the Groups containing Functional Component Definitions link in the Application Mapping channel, or from the group landing page. To export mapping definitions, you must have a minimum of one functional component defined.
To export application mapping definitions

1. From the Group page and with the Application Mapping tab selected, choose Actions > Export Application Mapping Definitions.

This screen illustrates the menu selection for exporting application mapping definitions.

2. Save the file to be exported and click OK.

⚠️ Importing existing mapping definitions

Importing an existing mapping definition set does not overwrite the original; a new set is created. You do not need to delete the mapping definition set after exporting it.

To import an exported mapping definitions file

You can restore to the original state using the import function. The group that you import is renamed, enabling you to clone your application mapping group. Collaborative Application Mapping (CAM) definitions cannot be imported into an earlier version than the one from which they were exported.

1. On the Administration tab, click Application Mapping Import in the Model section.

This screen illustrates the Application Mapping Import icon.

2. On the Application Mapping Import page, click Browse and select a file to import.
3. Click **Apply** to import the file.
   A confirmation message and icon display at the top of the page, with a link that enables you to quickly import another file. Additionally, the Results area displays the results of the import, listing the groups that were imported and the associated functional components that were created.

   This screen illustrates the confirmation message and results that display after importing an exported application mapping file.
   If the file you selected was previously imported, the application name is appended with the next available number to represent a copy of the same application saved as a different name.

4. To import another file, click the **New Import link** under the confirmation message.

**To export multiple mapping definitions**

1. From the Application Mapping channel on the **Applications** tab, click the **Groups containing Functional Component Definitions** link in the Other Actions section.
2. Select one or more groups that contain functional component definitions to export with mapping definitions.
3. From the **Actions** menu, select **Export Application Mapping Definitions**.

   This screen illustrates how to export Functional Component Definitions to export with mapping definitions.

4. Save the file to be exported and click **OK**.
5. To import the file, in the Model section of the **Administration** tab, click **Application Mapping Import**.

**Saving or editing the CAM model**

Making a change to an application mapping definition (the CAM model) increases the major version of the generated The Pattern Language (TPL). A major version change means that a new model is built in the datastore with the new patterns maintaining it. However, the old patterns are retained and they still maintain the old model. You cannot undo large changes to an application mapping definition, so before making changes you should create a backup of the application mapping definition using **Export Application Mapping Definitions**.
To save the current CAM model

1. Export the application mapping definitions. From the Group page and with the Application Mapping tab selected, choose Actions > Export Application Mapping Definitions.
2. Save the file to be exported and click OK. You are prompted for confirmation.

To change the CAM model

1. Modify the application mapping definitions to suit your new requirements.
2. Click Generate TPL.
   BMC Atrium Discovery automatically activates the pattern module that has been generated and submits it into the Knowledge Management UI.
3. Rediscover all of the hosts on which the Business Application Instance (BAI) components run. Alternatively you can manually run the generated patterns.
4. Examine the new model to check that it meets your new requirements.

If the model meets your new requirements
Delete the pattern module generated from the old definitions. This removes the BAIs and Functional Components created by the patterns.

If the model does not meet your new requirements

1. Delete the pattern module generated from the new definitions. This removes the BAIs and Functional Components created by the patterns.
2. Delete the new definitions.
3. Import an exported mapping definitions file.
4. Rediscover all of the hosts on which the BAI components run. Alternatively you can manually run the generated patterns.
5. Examine the updated model.

Application mapping cookbook

This page is currently restricted to the ADDM all and BMC Software groups.

Creating SIs in a pattern

You can write a TPL pattern that creates a Software Instance. Usually, the pattern describes the process of SI modelling and might be based on the following non-exhaustive information:

- Ambiguous conditions that trigger the SI creation.
- Methods for distilling the information that is necessary for SI attributes and detail nodes population. The information can be obtained from the items that are already discovered and modeled in BMC Atrium Discovery.
• Relationships with other items in BMC Atrium Discovery. For example, SI components, hosts and network devices that the SI depends on, and the like.

The Pattern templates topic contains a reference of existing patterns for modeling some commonly deployed software. To create an SI that is not covered by existing patterns in BMC Atrium Discovery Technology Knowledge Updates, see the information on creating a pattern.

Creating a static application model

BMC Atrium Discovery enables you to define a simple application by manual grouping the hosts and software instances in a specific way provided below.

Before you begin

Static application modeling became available in the Technology Knowledge Update 2014-Aug-1. Ensure you have the latest Technology Knowledge Update uploaded and activated.

To create a static application model

1. Open Applications from the top menu.
2. In the Application Mapping area, create a group that starts with "StaticApp: ", in exactly that form.

⚠️ Note

Do not create sub-groups, as they are not supported in a static application model.

3. Search for the Hosts, SoftwareInstances and LoadBalancerServices that are a part of the application you model and add them to the group you just created (in our example it is StaticApp:My Simple Application group).

⚠️ Note

During the CMDB sync, the static application model defined this way causes the creation of a BMC_Application CI in BMC Atrium CMDB, named after the group.

For example, a group named “StaticApp: Example Application” results in a BMC_Application CI with name “Example Application”.

As with all groups, the StaticApp group may contain any node kind, but only Host, SoftwareInstance and LoadBalancerService nodes will actually be related to the BMC_Application CI in the CMDB.
Managing reporting

The following topics explain how to run and manage the reports that are included with BMC Atrium Discovery:

- Reporting basics
- Main reports page
- Application reports
- Infrastructure reports
- Discovery reports
- Context-sensitive node reports

**Tip**

A simple way of reaching the reports that you use most often is to use a pre-defined dashboard, or to create one of your own. See Using and customizing dashboards for more information.

Reporting basics

BMC Atrium Discovery includes a number of predefined reports. Reports operate in the same way as searches: when a report is selected, a query is run to find matching objects. The results are displayed in the form of a list page, from which details of individual objects can be selected.

All of these reports can be added to a dashboard. See Using and customizing dashboards for more information.

Reports can also be located using the search in the UI. Keywords entered there are matched against the titles and descriptions of the reports.

Reports can be accessed from a number of areas in the product.

Main Reports page

- Operational Integrity reports
- Change Control reports
- Software reports
- Data Centre Standardization charts
- Atrium Discovery Deployment reports
- Data Centre Standardization Software reports
- Data Centre Standardization reports
- Host Resiliency - Services reports
- Host Resiliency - Kernel reports
Main Applications page

- Application Change Impact — See Application Change Impact.

Main Infrastructure page

- Virtualization — See Virtualization.
- Infrastructure Reports — See Reporting on the Infrastructure.
- Infrastructure Host Reports — See Infrastructure Host Reports.
- Installed Software — See Installed Software.
- Database Reports — See Database Reports.

Discovery Reports page

- Discovery Dashboard Reports — See Discovery Reports.
- New Entities — See Discovery Reports.
- Model Health — See Discovery Reports.

Context-sensitive node reports

Context-sensitive node reports are available for many nodes, the following are the principle ones:

- Hosts — See Host Reports.
- Network Devices — See Switch Reports.
- Mainframes — See Mainframe Reports.
- MF Parts — See Mainframe Reports.
- Software Instances — See Software Instance Reports.
- Packages — See Package Reports.
- Patches — See Patch Reports.
- Subnets — See Software Instance Reports.

Main reports page

Select the Reports section from the primary navigation bar to display the groups of reports. Some of these reports can also be viewed from the main Applications and Infrastructure pages.

- To run a report
- Report results
- To run reports on nodes
- To use chart reports
- To download report usage analysis
To run a report

1. From the Reports section, click the report you want to run. Alternatively, the main Application, Infrastructure and Policy pages display the most commonly used reports applicable to these sections.

Some reports are run immediately. For those that can be configured, a page is displayed enabling you to specify the items you want to report on.

2. Choose the data you want to report on by specifying the parameter fields.
   - The search is not case-sensitive.
   - Where a drop-down list is provided, you can select a specific value or where available you can choose All to match all values.
   - For dates, you can choose a day and month from drop-down lists and enter a year (in yyyy format). You can also specify a time (hours and minutes).

3. Click Run.
   The results are summarized in the form of a list object page. For information about list object pages and the charting and filtering tools available, see Viewing summary list pages.

The reports are categorized according to the following list. Click a category to see a list and description of the reports:

- Operational Integrity reports
- Change Control reports
- Software reports
- Data Centre Standardization charts
- Atrium Discovery Deployment reports
- Data Centre Standardization Software reports
- Data Centre Standardization reports
- Host Resiliency - Services reports
- Host Resiliency - Kernel reports

Report results

- The returned list displays selected attributes and relationships of each object. Click an item in the list to access the View Object page which displays all of an object's relationships and attributes.
- You can export a report in CSV format.
To run reports on nodes
Where reports list nodes, each node has a check box next to it. You can select and deselect individual or multiple nodes of this type using these check boxes. There is also a select all check box in the heading row which enables you to select all nodes in the report. The select all check box selects all nodes on all pages of the report.

To use chart reports
There are a number of functions available when viewing chart reports.

To view details of a chart
When viewing a chart, you can drill down into any part of that chart to see the details for that particular section. For example, the Host Vendor Distribution report, see Host Vendor Distribution.

1. From the Data Centre Standardization Charts section on the Reports page, click the Host Vendor Distribution report.
   From the pie chart view you can change the chart type. To do this, click the change chart type link displayed beneath the chart. Available chart types are:
   • Pie Chart
   • Horizontal Bar Chart
   • Vertical Bar Chart
2. Choose the data slice, or bar if you are viewing a bar chart, that you want to view in detail and click that area of the chart to view the details for that area.
   From the details view you can choose to view specific information from the list of available data.

To print a chart report
When viewing a pie chart or a bar chart, you can print that report.

1. Select a report and right-click the chart.
2. Click Print Chart.

To rotate a pie chart
When viewing a pie chart, you can rotate the angle of the pie chart and the annotations for each data slice.

1. Select a pie chart and right-click the chart.
2. Click Enable Rotation and drag your mouse in a circular motion until the chart is displayed in the required position.

To view slices of a pie chart
When viewing a pie chart, you can also enable a slice of data to move away from the pie chart in order to highlight that section of data.

1. Select a pie chart and right-click the chart.
2. Click **Enable Slicing Movement** and click the section of data that you want to highlight (as illustrated in the following screen).

![Pie Chart](image)

3. Click the slice again to return it to the original pie chart position.

**To download report usage analysis**

You can download information containing details on report usage in your environment. The information enables you to view and analyze which reports have been run and how often. This function can help you to identify the most and least valuable reports used in your network over time.

1. Click the **Support Services** link in the Appliance section of the Administration page.
2. Select the **Reports Usage** check box in the Miscellaneous Files section.
3. In the Create Archive section of the page, specify a name for your usage report in the Name row, such as August 2011 Reports.
   
   You can also add a description in the Description row.
4. Click “Gather”.
   
   You can select the check box corresponding to any gathered data file and delete that file by clicking **Delete**. You can also see a read-only summary of each file by clicking **View** in the Summary column.
5. To download the report, click the corresponding link in the **Name** column.

   The output for this example is as follows:

```
Report Counter
Name, Counter
DataCentreStand.Reports.HostByVendorDist,2
DataCentreStand.Reports.UnixDist,1
Databases.Reports.MySQL,2
Databases.Reports.SQLServer,1
Resiliency.Report.WindowsServicePackDist,1

Context Report Counter
Name, Counter
Host__Observed Communicating Hosts,2

Chart Counter
Name, Count
DataCentreStand.Reports.HostByVendorPie,25
DataCentreStand.Reports.SoftwareByCategoryPie,4
```
The report is divided into three areas:

- Report Counter
- Context Report Counter
- Chart Counter
  For each one there is a unique ID. The report is downloaded in the form of a text file.

**Operational Integrity reports**

The Operational Integrity section on the main Reports page enables you to run a number of reports that detail the operational aspects of the network. For example, hosts with failed PSUs, and hosts requiring DNS identification. This group of reports are designed to help you to eliminate avoidable causes of downtime in your network.

**Hosts with One or More Failed PSUs**

This report displays details of hosts which have one or more power supply units which have failed.

**Hosts with a Single PSU**

This report displays details of hosts which only have a single power supply unit; therefore there is no redundant PSU.

**Hosts with any IP Address requiring DNS identification**

Shows a list of Host interfaces with IP Addresses that are not resolved by DNS.

**Hosts with all IP Address requiring DNS identification**

Shows a list of Hosts that have no interfaces with IP Addresses that are resolved by DNS.

**Change Control reports**

The Change Control section on the Reports page enables you to run a number of reports that can provide you to gain a clear overview of your change process. Where these reports are accessed from the reports drop-down list, they are pre-populated with some appropriate parameters. When accessed from this page the report fields are free text entry fields.

**Host Hardware Configuration Changes**

This report shows detected Host Hardware Configuration changes over a user selectable period of time. Attributes considered are those in the Hardware and Network section of a Host. By default the time is the last day.

**Host OS Configuration Changes**

This report shows detected Host OS Configuration changes over a user selectable period of time. Attributes considered are those in the Operating System section of a Host. By default the time is the last day.
Software reports

The Software reports section on the Reports page enables you to run a number of discovered software-related reports.

Software Inventory - Summary

This report displays a summary of the distribution of versions of a piece of software.

Software Inventory - Detail

This report displays details of where software is running, with version information.

Software Inventory - Version Discovery Failures

This report displays software instances where discovery of version information failed. Does not include the cases where no attempt was made to discover the version.

Runtime Environment Inventory - Summary

Shows a summary of the distribution of versions of a runtime environment.

Runtime Environment Inventory - Detail

Shows details of where runtime environments are running, with version information.

Runtime Environment Inventory - Version Discovery Failures

This report displays runtime environments where discovery of version information failed. Does not include the cases where no attempt was made to discover the version.

Data Centre Standardization charts

The Data Centre Standardization Charts section on the Reports page enables you to run a number of discovered data centre-related charts.

Host Vendor Distribution

This report shows a list of the Hosts per Vendor.

Windows Version Distribution

This report shows a list of the Version for Windows.

Windows Version and Edition Distribution

This report shows a list of the Version and Edition for Windows.

UNIX Type Distribution

This report shows a distribution of UNIX type operating systems (includes GNU/Linux and FreeBSD).
UNIX Version Distribution
This report shows a list of UNIX type operating systems (includes GNU/Linux and FreeBSD) and Version.

Atrium Discovery Deployment reports
The BMC Atrium Discovery Deployment section on the Reports page enables you to run a number of reports that detail deployment and data completeness information.

Hosts and other devices that failed last CMDB sync
Shows Hosts, Network Devices and other root device types that were not successfully synchronized with the CMDB on the most recent attempt.

Aged-Out Hosts and other devices that failed last CMDB sync
Shows Hosts, Network Devices and other root device types that are marked as destroyed, but were not successfully synchronized with the CMDB on the most recent attempt. These devices should be synchronized manually, to ensure they are not present in the CMDB.

BMC ADDM License Audit
Shows anonymized OSI count as requested by BMC License Operations.

BMC ADDM License Audit (non-anonymized)
Shows OSI count for the customer reference, without anonymizing.

Imported root node key information which has not been used
Shows key information for root nodes which has not been used during discovery to create new root nodes.

Existing pattern definitions for software mapping
Shows all existing pattern definitions for software mapping.

Data Centre Standardization Software reports
The Data Centre Standardization Software section on the Reports page enables you to run a number of discovered data center-related software.

Software Distribution by Category
This report shows a distribution of the Software by Category. Software that is in more than Category will be counted more than once.

Software Detail by Category
This report shows a detail list of the Software by Category. Software that is in more than Category will be displayed more than once.
Software Distribution by Category
This report shows a list of the Software by Category in the form of a pie chart. Software that is in more than one Category will be counted more than once.

Data Centre Standardization reports
The Data Centre Standardization reports section on the Reports page enables you to run a number of discovered data centre-related reports.

Host Distribution by Vendor
This report shows a distribution of the Hosts per Vendor. Unknown Vendors are excluded, for this detail use Host Distribution By Vendor and Model

Host Summary by Vendor
This report shows a list of the Hosts per Vendor.

Host Distribution by Vendor and Model
This report shows a distribution of the Hosts per Vendor and Model.

Windows Version Distribution
This report shows a distribution of Windows versions.

Windows Version and Edition Distribution

UNIX Type Distribution
This report shows a distribution of UNIX type operating systems (includes GNU/Linux and FreeBSD).

UNIX Version Distribution
This report shows a distribution of UNIX type operating systems (includes GNU/Linux and FreeBSD) and version.

Host Resiliency - Services reports
The Host Resiliency - Services section on the Reports page enables you to run a number of discovered services-related host reports.

Summary of Hosts with Service Port Open
This report shows a summary of which Hosts have been seen with common service ports open. The report can be restricted to just the latest scan data for each Host, or all available scan data.
Detail of Hosts with service port open
This report shows the detail of which Hosts have been seen with common service ports open. The report runs over all available scan data and shows the port details, the number of connections seen, the time it was seen and basic host details.

Host Resiliency - Kernel reports
The Host Resiliency - Kernel section on the Reports page enables you to run a number of discovered kernel-related host reports.

Windows Service Pack Distribution
This report shows a distribution of Windows Service Packs.

Find Windows Hosts Missing a Service Pack
This report shows a list of Windows Hosts missing a Service Pack. You will need to provide the Windows Version and Service Pack number. The report will only consider hosts of that Windows Version.

Find Windows Hosts without Minimum Service Pack
This report shows a list of Windows Hosts with a Service Pack less than the one specified. You will need to provide the Windows Version and Service Pack number. The report will only consider hosts of that Windows Version.

UNIX Kernel Distribution
This report shows a distribution of UNIX Kernels. Includes FreeBSD and GNU/Linux.

Application reports
The main Applications page enables you to run a number of reports on the Business Applications in your system. These individual reports can also be run from the main Reports home page.

- **Applications that could be Impacted by a Network Device** — This report shows a list of Applications that could be impacted by changes to a Network Device. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs.

- **Applications that could be Impacted by a Network Device via Drop Down** — This report shows a list of Applications that could be impacted by changes to a Network Device. This report uses a drop down of all seen Network Devices. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs.

- **Applications that could be impacted by a Host** — This report shows a list of Applications that could be impacted by changes to a Host. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this Host.
• **Applications that could be Impacted by a Host via Drop Down** — This report shows a list of Applications that could be impacted by changes to a Host. This report uses a drop down of all seen Hosts. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this Host.

• **Applications that could be Impacted by a Host Model** — This report shows a list of Applications that could be impacted by changes to a Host Model. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this Host Model.

• **Applications that could be Impacted by a Host Model via Drop Down** — This report shows a list of Applications that could be impacted by changes to a Host Model. This report uses a drop down of all seen Host Models. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this Host Model.

• **Applications that could be Impacted by an OS** — This report shows a list of Applications that could be impacted by changes to an OS. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this OS.

• **Applications that could be Impacted by an OS via Drop Down** — This report shows a list of Applications that could be impacted by changes to an OS. This report uses a drop down of all seen OSs. Set Include Hosted Virtual OSIs to also Include Applications running in Virtual OSIs hosted by this OS.

• **Applications that could be Impacted by a Software Product** — This report shows a list of Applications that could be impacted by changes to an Software Instance type. By default this report only searches directly modeled containment dependencies. Set Include Co Hosted Applications to include Applications that run on the same Host. Set Include Co Inferred Software Instances to include Software Instances with primary inferences from the same source, for instance this would include Java SIs when searching for Tomcat SIs. Both these additions will cause the report to execute slower.

• **Applications that could be Impacted by a Software Product via Drop Down** — This report shows a list of Applications that could be impacted by changes to an Software Instance type. This report uses a drop down of all seen Software Instance Types. By default this report only searches directly modeled containment dependencies. Set Include Co Hosted Applications to include Applications that run on the same Host. Set Include Co Inferred Software Instances to include Software Instances with primary inferences from the same source, for instance this would include Java SIs when searching for Tomcat SIs. Both these additions will cause the report to execute slower.

### Infrastructure reports

The main Infrastructure page enables you to run a number of reports on the IT infrastructure. Some of these individual reports can also be run from the main Reports home page.
Virtualization and Partitioning General Reports

- **All Contained Hosts**— Shows a report of Hosts that are Virtual (for example, VMware, Solaris Zones), a member of a Cluster (for example, Veritas Cluster Server) or a member of a Host Container (for example, Sun Enterprise 15K Series).
- **All Partition Hosts** — Shows a report of Hosts that are Partitions (for example, Sun Enterprise or IBM POWER).
- **All Virtual Hosts** — Shows a report of Hosts that are Virtual (for example, VMware, Solaris Zones).
- **Containing Hosts with Virtual Machines** — Shows a report of Hosts that are running Virtual Machines and how many have been linked to virtual Hosts.
- **Virtual Hosts relying on Containing Hosts** — Shows a report of Hosts that are Virtual Machines (for example, VMware, Solaris Zones, HyperV) and their corresponding Containing Hosts.
- **Virtual Machines not linked to Containing Hosts** — Shows a report of Hosts that appear to be running in Virtual Machines but have not been linked to containing Hosts.
- **VM Technology Classification** — Shows a count of virtual Hosts for each virtualization technology.

VMware Infrastructure

- **Hosts supporting VMware VMs** — Shows a report of the physical Hosts containing running VMware Guests.
- **ESX/ESXi Hosts where API access needs patching** — Shows a report of the ESX/ESXi Hosts that have a connection leak in their API and need updating if API based discovery is required.
- **Hosts in VMware VMs** — Shows a report of the contained VMware Guests.
- **VMware Guest OS Classification** — Shows a chart of OS Classification of VMware Guests.
- **VMware Guest OS Distribution** — Shows a chart of OS Type distribution of VMware Guests.

Microsoft Hyper V Infrastructure

- **Hosts supporting Microsoft Hyper-V VMs** — Shows a report of the Hosts containing Hyper-V hosts.
- **Hosts in Microsoft Hyper-V VMs** — Shows a report of the contained Microsoft Hyper-V hosts.
- **Microsoft Hyper-V Guest OS Classification** — Shows a chart of OS Classification of Microsoft Hyper-V Guests.
- **Microsoft Hyper-V Guest OS Distribution** — Shows a chart of OS Type distribution of Microsoft Hyper-V Guests.
IBM POWER Systems Infrastructure

- **IBM POWER HMCs** — Shows a report of Hosts that are IBM POWER HMCs and what systems they are configured to manage.
- **IBM POWER Frames** — Shows a report of Host Containers that are IBM POWER system frames.
- **IBM POWER LPARs** — Shows a report of Hosts that are LPARS on IBM POWER systems. Includes AIX, Linux and VIO Hosts.
- **AIX LPARs supporting WPAR VMs** — Shows a report of the LPARs containing WPARs.
- **Hosts in AIX WPAR VMs** — Shows a report of the contained AIX WPARs.
- **Power LPAR/WPAR OS Distribution** — Shows a chart of OS Type distribution of IBM POWER LPAR/WPAR Hosts.
- **Power LPAR/WPAR OS Classification** — Shows a chart of OS Classification of IBM POWER LPAR/WPAR Hosts.

Solaris Infrastructure

- **Solaris LDOMs** — Shows a report of Hosts that are Solaris Logical Domains.
- **Hosts supporting Solaris Zones** — Shows a report of the physical Hosts containing Solaris Zones.
- **Hosts in Solaris Zones** — Shows a report of the contained Solaris Zones

Other Virtualization Technologies

- **Hosts supporting Xen VMs** — Shows a report of the Hosts containing Xen Domain hosts.
- **Hosts in Xen VMs** — Shows a report of the contained Xen Domain hosts.
- **Hosts supporting OracleVM VMs** — Shows a report of the Hosts containing OracleVM Domain hosts.
- **Hosts in OracleVM VMs** — Shows a report of the contained OracleVM Domain hosts.
- **Hosts supporting OracleVM VirtualBox VMs** — Shows a report of the physical Hosts containing running OracleVM VirtualBox Guests.
- **Hosts in OracleVM VirtualBox VMs** — Shows a report of the contained OracleVM VirtualBox Guests.

File System Reports

- **Full File Systems** — Shows Hosts with very full File Systems.
- **Free Space** — Shows Hosts with File Systems which have a lot of free space.

Mainframe Software Reports

- **Software on all LPARs** — Shows All All Software on all LPARs.
• All Software on a specific LPAR — Shows list of All Software on a specific LPAR by dropdown. This uses a drop down of all LPARs discovered.
• Mainframe Transaction Servers — Shows Mainframe Transaction Servers.
• Detailed CICS Transactions — Shows Detailed CICS info.
• Detailed CICS Transaction Programs — Shows Detailed CICS Programs
• Detailed IMS Transactions — Shows Detailed IMS info.
• Mainframe Database Servers — Shows Mainframe Database Servers.
• Databases in IMS — Shows Databases in IMS.
• Databases in DB2 — Shows Databases in DB2.
• DB2 Details per Database — Shows DB2 indexes, tables, pagesets per Database.
• MQ Queue Managers — Shows MQ Queue Managers.
• MQ Details — Shows MQ channels, listeners, queues.
• Websphere — Shows Websphere.

Host Storage Reports

• Host Storage — Show Host Storage.
• SAN Storage Volumes used by Hosts — Show SAN Storage Volumes in use by Hosts.
• Physical Disks used by Hosts — Show Physical Disks in use by Hosts.
• Host Virtual Disks & Physical Storage — Show Host Virtual Disks and the backing Physical Storage.

Operating Systems

• Host OS Classification — Shows a count of Hosts for each OS Classification.
• Host OS Distribution — Shows a count of Hosts for each OS Type.
• UNIX Classification — Shows a count of Hosts for each UNIX OS.
• Windows Versions — Shows a count of Hosts for each Windows Version.

Infrastructure Reports

• Hosts with Organizational Details — Shows a list of Hosts, with details of linked owners, locations, and organizational units.
• Host NIS/Windows Domain — Show a list of Host nodes and their discovered domain ordered by domain and hostname.
• Host NIS/Windows Domain Chart — Show a chart of Host nodes and their discovered domain.
• Fibre Channel HBA — Shows a list of Fibre Channel Host Based Adapters (HBA) and which Host they are installed in. The list includes the HBA identifier, WWNN and WWPN.
• Network Interface MAC to IP — Shows a list of Network Interface MAC addresses against the IP addresses allocated to them and which Host they are installed in. The list includes the Interface Name, MAC and IP. Virtual interfaces on the same MAC address are shown on separate rows.
• **Host/Network Device Mismatches** — Shows Host to Network Device configuration mismatches.
• **Host IP Address Distribution** — Shows distribution of IP Addresses per Host.
• **All Blade Enclosures** — Shows all Blade Enclosures.
• **Hosts with Management Controllers** — Shows all Hosts managed by a Management Controller.

### Load Balancer Reports

• **All Load Balancer Devices** — Shows all Network Devices with load balancer capabilities.
• **Load Balancers in failover pairs** — Shows Load Balancers configured in failover pairs.
• **Load Balancer Members not linked to a Host** — Shows Load Balancer Members that are not linked to a Host.
• **Load Balancer Members not linked to software** — Shows Load Balancer Members that are not linked to a Software Component or Software Instance.
• **Load Balancer Services missing links to software** — Shows Load Balancer Services that use members which are not linked to a Software Component or Software Instance.

### Installed Software

• **Installed Packages** — Report to show Hosts linked to Packages. This is an intensive report and might take some time to run.
• **Packages - Where Installed** — Shows package details and on what hosts they are installed. List can be filtered by package name.
• **Patches - Where Installed** — Shows patch details and on what hosts they are installed. List can be filtered by patch name.

### Infrastructure Host Reports

• **Seen but unscanned IPs Summary Report** — This report displays summary details of Hosts that were seen but are not present in the datastore. That is, a Host that was not discovered, for example, it was behind a firewall, on a subnet that was not scanned, or credentials were not available to log in to it. However, its existence can be inferred by network connections to its IP address from known hosts.
• **Seen but unscanned IPs Detail Report** — This report displays details of Hosts that were seen but not scanned. That is, a Host that was not discovered, for example, it was behind a firewall or on a subnet that was not scanned. However, its existence can be inferred by network connections to its IP address from known hosts.
• **Newly Discovered Hosts** — Report to show new Hosts found in the given period.

### Database Reports

• **All Databases Instances** — Show a list of all Databases Instances.
• **Oracle Instances** — Show a list of Oracle Instances.
- **Sybase Instances** — Show a list of Sybase Instances.
- **SQL Server Instances** — Show a list of SQL Server Instances.
- **MySQL Instances** — Show a list of MySQL Instances.
- **DB2 Instances** — Show a list of DB2 Instances.
- **PostgreSQL Instances** — Show a list of PostgreSQL Instances.
- **Informix Instances** — Show a list of Informix Instances.

**Mainframe Infrastructure Reports**

- **All Mainframes** — Shows All mainframes.
- **Sysplexes** — Shows Sysplex clusters.
- **All Mainframe LPARs** — Shows All mainframe LPARs.
- **All LPARs on a specific Mainframe** — Shows list of All LPARS on a specific Mainframe by dropdown. This uses a drop down of all Mainframes discovered.
- **Virtual LPAR to Native LPAR map** — Shows Virtual LPAR to Native LPAR map.
- **Mainframe Storage Subsystems** — Shows Mainframe Storage Subsystems.
- **Mainframe Storage Volumes** — Shows Mainframe Storage Volumes.
- **Coupling Facility** — Shows Coupling Facilities.

**NAS Storage Reports**

- **All NAS Storage Systems** — Shows all NAS Storage Systems.
- **Hosts connected to NAS Storage Systems** — Shows all Hosts mounting filesystems from NAS Storage Systems.
- **Applications running on Hosts connected to NAS Storage** — Show Applications running on Hosts that are connected to a NAS Storage System.
- **NAS Storage Systems by Vendor** — Shows a count of NAS Storage Systems for each Vendor.

**SAN Storage Reports**

- **All SAN Storage Systems** — Shows all SAN Storage Systems
- **All Storage Volumes** — Shows all Storage Volumes
- **Storage Management Software** — Shows Storage Management software details.
- **Connections Not Matched With A Host** — Shows Storage Connections that are not linked to a host. This may be because the Host has not been discovered yet or no longer exists.
- **SAN Storage Volumes used by Hosts** — Show SAN Storage Volumes in use by Hosts.
- **Hosts with HBA Cards Not Connected to Storage** — Shows Hosts that have HBA cards that are not mapped to any storage. This may be because they are connected to Storage Systems that have not been discovered, or because the HBA cards are no longer used.
- **Hosts Connected to Storage (SAN)** — Show Hosts that are connected to a SAN Storage System.
- **Applications running on Hosts connected to SAN Storage** — Show Applications running on Hosts that are connected to a SAN Storage System.
- **Storage Pools with an oversubscription warning** — Shows Storage Pool where the Oversubscription warning has been set.
- **SAN Storage Systems by Vendor** — Shows a count of SAN StorageSystems for each Vendor.

### Discovery reports

When you click the **Discovery** tab of the primary navigation bar, the main Discovery Reports page can be accessed by clicking the Discovery Reports button in the dynamic secondary navigation bar. This page enables you to run a number of reports on Discovery Access.

### Discovery Dashboard reports

- **Current Endpoint Access Analysis** — Allows the analysis of the latest endpoint access to assist Discovery.
- **Host Endpoints Not Updating** — List of Endpoints related to existing Host nodes that do not cause a Host node update on their last access.
- **Host Endpoints With Session Issues** — List of Endpoints related to existing Host nodes that caused a Host node update on their last access, but also had Session Issues.
- **Host Endpoints With Possible Process To Port Issues** — List of Endpoints related to existing Host nodes that caused a Host node update on their last access, but for which we cannot determine Process to Port mappings.
- **Possible Endpoint Host Devices (Detailed)** — List of Endpoints not accessed or related to current Hosts but believed to be Host devices on their last access, with detailed OS breakdown where available.
- **Possible Endpoint Non Host Devices** — List of Endpoints not accessed or related to current Hosts but believed to be non Host devices such as printers, routers and so on on their last access.
- **View Current Discovery Runs** — View Current Discovery Runs.
- **View Current Consolidation Runs** — View Current Consolidation Runs.
- **Seen but unscanned IPs** — List of IPs that we have not scanned for which connections were seen in the last access.
- **Seen but unscanned IPs with Ports** — List of IPs that we have not scanned for which connections were seen in the last access. This report lists which port the connection was to, but will result in a much more detailed report. Initially use the variant of this report with no port information.
- **Summary of Last Endpoint Access for Host Devices** — This chart shows a summary of the access used on the last access to the endpoints classified as a Host.
- **Blade Enclosure Discovery Summary** — Shows a summary of the discovery of each Blade Enclosure. Note that some vendors, such as Dell, do not report information on I/O Modules so the expected value can be zero.
- **Undiscovered Hosts for Processor Blades** — Shows Processor Blades without a related Host.
• **Undiscovered Network Devices for I/O Modules** — Shows I/O Modules without a related Network Device. Note: some vendors, such as Dell, don't report I/O module information.

• **Management Controllers without Hosts** — Shows all Management Controllers which are not connected to a Host

• **Discovery Conditions Detailed List** — Shows a list of Discovery Conditions logged, including time first logged. Use this report to export a tasklist via CSV.

• **Hosts Impacted by Discovery Conditions** — Shows a list of Hosts impacted by selected Discovery Conditions.

• **Imported root node key information which has not been used** — Shows information for root nodes whose keys have been imported but have not yet been used.

**New entities**

• **Newly Discovered Hosts** — Report to show new Hosts found in the given period.

• **Newly Discovered Software** — Report to show new Software Instances found in the given period.

• **Newly Discovered Packages** — Report to show new Packages found in the environment in the given period. This is an intensive report and might take some time to run.

• **New Packages discovered on Hosts** — Report to show new links between Hosts and Packages in the given period. This is an intensive report and might take some time to run.

• **Newly Discovered Patches** — Report to show new Patches found in the environment in the given period. This is an intensive report and might take some time to run.

• **New Patches discovered on Hosts** — Report to show new links between Hosts and Patches in the given period. This is an intensive report and might take some time to run.

• **Installed Packages** — Report to show Hosts linked to Packages. This is an intensive report and might take some time to run.

**Model health**

• **Devices with Endpoint Identity Change** — Show a list of root inferred nodes created when the identity of a previously existing node changed. The previous node is no longer actively being maintained and is aging out.

• **Hosts near Removal Threshold** — Show a list of Host nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **Network Devices near Removal Threshold** — Show a list of Network Device nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **MFP Parts near Removal Threshold** — Show a list of MFP Part nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **Printers near Removal Threshold** — Show a list of Printer nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **SNMP Managed Devices near Removal Threshold** — Show a list of SNMP Managed Device nodes that are near their removal thresholds and gives an approximate indication of the removal deadline
• **Management Controllers near Removal Threshold** — Show a list of Management Controller nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **Storage Devices near Removal Threshold** — Show a list of Storage Device nodes that are near their removal thresholds and gives an approximate indication of the removal deadline

• **Hosts impacted By Discovery Conditions** — Shows a list of Hosts impacted by selected Discovery Conditions

• **Containing Hosts with Virtual Machines** — Shows a report of Hosts that are running VMs and how many have been linked to virtual Hosts

• **Virtual Machines not linked to Containing Hosts** — Shows a report of Hosts that appear to be running in Virtual Machines but have not been linked to containing Hosts

**Context-sensitive node reports**

BMC Atrium Discovery enables you to run many different kinds of reports on some of the nodes in your system. Some of these individual reports can also be run from other areas of the product.

**Host reports**

To run host-related reports:

1. Click the Hosts link in the Infrastructure Summary section of the Infrastructure page. The Hosts page is displayed.
2. Click the **Reports** drop down.

   The following host-related reports are displayed:

   • **Seen but unscanned IPs** — Show unscanned Hosts inferred from Network Connections.
   • **Network Device Dependencies** — Show Network Devices used by these Hosts.
   • **Host Dependencies via SI and BAI** — Show other Hosts via Traversing shared SI and BAI.
   • **Number of Interfaces vs Hosts** — Show the number of Network Interfaces versus Number of Hosts.
   • **Number of Subnets vs Hosts** — Show the number of Subnets versus Number of Hosts.
   • **Observed Communicating Hosts** — Show hosts reached through observed network connections.
   • **Detailed Observed Communications** — Detailed report of observed communication showing processes where available.
   • **Storage Systems** — Show Storage Systems these Hosts may access.
   • **Storage Volumes** — Show Storage Volumes these Hosts may access.
   • **Host Last Update Summary** — Show the distribution of the relative age of the last update on Hosts.
- **Number of Interfaces vs Hosts** — Show the number of Network Interfaces versus Number of Hosts on a Column Chart.
- **Number of Subnets vs Hosts** — Show the number of Subnets versus Number of Hosts on a Column Chart.
- **Number of Software Instances vs Hosts** — Show the number of Software Instances running on a Host versus Number of Hosts on a Column Chart.
- **Number of Applications vs Hosts** — Show the number of Business Application Instances running on a Host versus Number of Hosts on a Column Chart.
- **Host OS Type Summary Chart** — Show OS by Type on a Pie Chart.
- **Host OS Class Summary Chart** — Show OS by Class on a Pie Chart.
- **Host Access Method Summary Chart** — Show Host Access Methods on a Column Chart.
- **Host Access Result Summary Chart** — Show Host Access Results on a Column Chart.

**Individual host reports**

To run reports on individual Hosts in your system:

1. Click the Hosts link in the Infrastructure Summary section of the Infrastructure page.
2. Select a host from the list.
3. Click the Reports drop down.

The following reports are available on each host:

- **Communication Summary** — Summary of the observed communication.
- **Observed Communicating Hosts** — Hosts reached through observed network connections.
- **Current Network Connections** — Network Connections from last scan.
- **Historical Network Connections** — Line Graph of the Number of Network Connections from all available scans.
- **Current Open Ports** — Open Ports from last scan.
- **Historical Open Ports** — Open Ports from all available scans.
- **Current Processes** — Processes from last scan.
- **Historical Processes** — Processes from all available scans.
- **Current Services** — Services from last scan.
- **Historical Services** — Services from all available scans.
- **Seen but unscanned IPs** — Show unscanned Hosts inferred from Network Connections.
- **Network Device Dependencies** — Show Network Devices used by this Host.
- **Host Dependencies via SI and BAI** — Show other Hosts via Traversing shared SI and BAI.
- **Detailed Observed Communications** — Detailed report of observed communication showing processes where available.
- **Storage Systems** — Show Storage Systems this Host may access.
- **Storage Volumes** — Show Storage Volumes this Host may access.
• **Host Network Connection Count Over Time** — Line Graph of the Number of Network Connections over all available scans.
• **Host Process Count Over Time** — Line Graph of the Number of Processes over all available scans.
• **Host Access Method Summary Chart** — Host Access Methods on a Column Chart.
• **Host Access Result Summary Chart** — Host Access Results on a Column Chart.

**Network Device reports**

To run Network Device-related reports:

1. Click the Network Devices link in the Infrastructure Summary section of the Infrastructure page.
2. Click the **Reports** drop down.
   
   The following Network Device-related report is displayed:
   
   • **Connected Hosts** — Number of Hosts connected to these Network Devices.

**Individual Network Device reports**

To run reports on individual Network Devices in your system:

1. Click the Network Devices link in the Infrastructure Summary section of the Infrastructure page. The Network Devices page is displayed.
2. Select a Network Device from the list.
3. Click the **Reports** drop down.
   
   The following reports are available for a Network Device:
   
   • **Hosts connected to Network Device `<Network Device Name>`** — Hosts connected to this Network Device.
   • **Subnets connected to Network Device `<Network Device Name>`** — Subnets connected to this Network Device.
   • **Interface Details for Network Device `<Network Device Name>`** — Interface details of this Network Device.
   • **Speed of Interface connected to Network Device `<Network Device Name>`** — The speed of the Network Device Interface, displayed on a pie chart.
   • **Duplex of Interfaces connected to Network Device `<Network Device Name>`** — The duplex of the Network Device Interface, displayed on a pie chart.
   • **Host OS Type connected to Network Device `<Network Device Name>`** — The OS of the connected host by type, displayed on a pie chart.
   • **Host OS Class connected to Network Device `<Network Device Name>`** — The OS of connected host by class, displayed on a pie chart.

**Management Controller reports**

**Individual Management Controller reports**

To run reports on individual Management Controllers in your system:
1. Click the Management Controller link in the Infrastructure Summary section of the Infrastructure page. The Management Controllers page is displayed.
2. Select a Management Controller from the list.
3. Click the Reports drop down.

   The following reports are available for a Network Device:
   - **Subnets connected to device <Device Name>** — Subnets Connected to this Management Controller.

**Mainframe reports**

To run Mainframe-related reports:

1. Click the Mainframe link in the Infrastructure Summary section of the Infrastructure page.
   The Mainframe page is displayed.
2. Click the Reports drop down.

   The following Mainframe-related reports are available:
   - **Contained MFParts** — LPARs on these Mainframes.
   - **Contained Sysplex** — Sysplex on these Mainframes.
   - **Contained Coupling Facilities** — Coupling Facilities on these Mainframes.
   - **Contained z/VM Guest LPARs** — LPARs running under z/VM mapped to the z/VM LPARs on these Mainframes.
   - **Storage SubSystems** — Storage SubSystems used by LPARs on these Mainframes.
   - **Storage Volumes** — Storage Volumes used by LPARs on these Mainframes.

**Individual Mainframe reports**

To run reports on individual Mainframe in your system:

1. Click the Mainframe link in the Infrastructure Summary section of the Infrastructure page.
2. Select a Mainframe from the list.
3. Click the Reports drop down.

   The following reports are available for the Mainframe:
   - **Contained MFParts** — LPARs on this Mainframe.
   - **Contained Sysplex** — Sysplex on these Mainframes.
   - **Contained Coupling Facilities** — Coupling Facilities on these Mainframes.
   - **Contained z/VM Guest LPARs** — LPARs running under z/VM mapped to the z/VM LPARs on this Mainframe.
   - **Storage Sub-systems** — Storage SubSystems used by LPARs on this Mainframe.
   - **Storage Volumes** — Storage Volumes used by LPARs on this Mainframe.

**MFPart reports**

To run MFPart-related reports:
1. Click the MFPart link in the Infrastructure Summary section of the Infrastructure page. The MFPart page is displayed.
2. Click the Reports drop down.
   The following MFPart-related reports are available:
   - **Contained in Sysplex** — Sysplex these LPARs are part of.
   - **Coupling Facilities** — Coupling Facilities used by these LPARs.
   - **Storage SubSystems** — Storage SubSystems used by these LPARs.
   - **Storage Volumes** — Storage Volumes used by LPARs.
   - **Containing Mainframe** — Mainframe that contains these LPARs.
   - **Peer LPARs via Sysplex** — Other LPARs in the same Sysplex as this one.
   - **Peer LPARs via Coupling Facility** — Other LPARs using the same Coupling Facility as this one.
   - **Peer LPARs via Storage SubSystem** — Other LPARs using the same Storage SubSystems as this one.

**Individual MFPart reports**

To run reports on individual MFPart in your system:

1. Click the MFPart link in the Infrastructure Summary section of the Infrastructure page.
2. Select a MFPart from the list.
3. Click the Reports drop down.
   The following reports are available for the MFPart:
   - **Containing Mainframe** — Mainframe that contains these LPARs.
   - **Peer LPARs via Sysplex** — Other LPARs in the same Sysplex as this one.
   - **Peer LPARs via Coupling Facility** — Other LPARs using the same Coupling Facility as this one.
   - **Peer LPARs via Storage SubSystem** — Other LPARs using the same Storage SubSystems as this one.

**Software instance reports**

To run SI-related reports:

1. Click the Software Instances link in the Infrastructure Summary section of the Infrastructure page.
2. Click the Reports drop down.
   The following SI-related reports are available:
   - **Host Dependencies** — Shows hosts used by these software instances.
   - **Detailed Observed Communication** — Shows a summary of the communication observed for these software instances.
   - **Software Instance Type Summary** — Shows a summary of the software instance types.
   - **Application Summary** — Shows a summary of the applications using these software instances types.
Individual software instance reports

To run reports on individual software instances in your system:

1. Click the Software Instances link in the Infrastructure Summary section of the Infrastructure page.
2. Select a software instance from the list.
3. Click the Reports drop down.

The following reports for this software instance are available:

- **Other <SI Type> <SI Version> Instances** — Shows other software instances that have this same full version.
- **Other <SI Type> <Product Version> Instances** — Shows other software instances that have this same product version.
- **Other <SI Type> Instances** — Shows other software instances that are the same type.
- **Other Items maintained by this Pattern** — Shows other items that are maintained by the same pattern.
- **Hosts with <SI Type> <SI Version> Instances** — Shows hosts that have this same full version.
- **Hosts with <SI Type> <Product Version> Instances** — Shows hosts that have this same product version.
- **Hosts with <SI Type> Instances** — Shows hosts that have this same type.
- **Version Distribution** — Shows the versions of this type.
- **Detailed Observed Communication** — Shows a summary of the communication observed for these software instances.
- **Possible Version Failures** — Shows software instances of this type with no version derived.
- **Possible Version Failure Hosts** — Shows hosts with software instances of this type with no version derived.
- **Storage Systems** — Storage Systems that this SI may access.
- **Storage Volumes** — Storage Volumes that this SI may access.
- **Full Version Distribution for <SI Type>** — Displays the full versions of this type on a pie chart.
- **Product Version Distribution <SI Type>** — Displays the product versions of this type on a pie chart.

Business Application Instance reports

A number of Business Application Instance (BAI)-related reports are available on the Application Instances page.

To access BAI reports:

1. Click the Application Instances link in the Application Summary section of the Applications page.
2. Click the **Reports** drop down.

   The following BAI-related reports are available:
   - **Host Dependencies** — Shows hosts used by these BAIs.
   - **OS Dependencies** — Shows the OS used by these BAIs.
   - **Software Dependencies** — Shows the software instances used by these BAIs.
   - **Network Device Dependencies** — Shows Network Devices used by these BAIs.
   - **Business Application Type Summary** — Shows a summary of the BAI types.
   - **Software Instance Summary** — Shows a summary of the Software Instance being used by these applications.

**Individual application instance reports**

To run reports on individual application instances in your system:

1. Click the Application Instances link in the Application Summary section of the Applications page. The Application Instances page is displayed.
2. Select an application instance from the list.
3. Click the **Reports** drop down.

   The following reports for the software instance are available:
   - **Host Dependencies for <BAI Name>** — Shows hosts used by this BAI.
   - **OS Dependencies for <BAI Name>** — Shows the OS used by this BAI.
   - **Software Dependencies for <BAI Name>** — Shows software instances used by this BAI.
   - **Network Device Dependencies for <BAI Name>** — Shows Network Devices used by this BAI.
   - **Other <BAI Type> Instances** — Shows other BAIs that are the same type.
   - **Other Items maintained by this Pattern** — Shows other items that are maintained by the same pattern.
   - **Hosts with <BAI Type> Instances** — Shows hosts that have this same type.
   - **Version Distribution** — Shows the versions of this type.
   - **Possible Version Failures** — Shows BAIs of this type with no version derived.
   - **Possible Version Failure Hosts** — Shows hosts with BAIs of this type with no version derived.
   - **Full Version Distribution for <BAI Type>** — Displays the full versions of this type on a pie chart.
   - **Product Version Distribution for <BAI Type>** — Displays the product versions of this type on a pie chart.

**Package reports**

To run package-related reports:

1. Click the Package link in the Infrastructure Summary section of the Infrastructure page. The Package page is displayed.
2. Click the **Reports** drop down.  
The following package-related reports are available:  
   - **Package Host Density** — How many hosts are these packages installed on.

### Individual package reports

To run reports on individual packages in your system:

1. Click the Packages link in the Infrastructure Summary section of the Infrastructure page.  
2. Select a package from the list.  
3. Click the **Reports** drop down.  
The following reports are available for the package:  
   - **Hosts with Package Name** — Shows other hosts with this named package on.  
   - **Versions of Package Name** — Show the versions of this package.  
   - **Detail Versions of Package Name** — Show the versions and revisions of this package.

### Patch reports

To run patch-related reports:

1. Click the Patch link in the Infrastructure Summary section of the Infrastructure page. The Patch page is displayed.  
2. Click the **Reports** drop down.  
The following Patch-related reports are available:  
   - **Patch Host Density** — How many hosts are these patches installed on.

### Individual patch reports

To run reports on individual patches in your system:

1. Click the Patch link in the Infrastructure Summary section of the Infrastructure page.  
2. Select a patch from the list.  
3. Click the **Reports** drop down.  
The following reports are available for the patch:  
   - **<OS-based> Hosts with Patch** — Shows other hosts with the same Operation System that have this patch on.

### Subnet reports

To run subnet-related reports:

1. Click the Subnets link in the Infrastructure Summary section of the Infrastructure page. The Subnets page is displayed.  
2. Click the **Reports** drop down.  
The following subnet-related reports are available:  
   - **Subnet Host Count** — How many hosts are on these subnets.  
   - **Subnet Device Count** — How many Devices are on these subnets.
Individual subnet reports
To run reports on individual subnets in your system:

1. Click the Subnets link in the Infrastructure Summary section of the Infrastructure page.
2. Click the Reports drop down.
3. Click the Reports drop down.
   The following reports are available for the subnet:
   - **Hosts on this Subnet** — Shows other hosts that have an interface on this subnet.
   - **Devices on this Subnet** — Shows other interfaces on this subnet.
   - **OS Class on this Subnet** — Displays the OS Class of the hosts on this subnet in a pie chart.
   - **OS Type on this Subnet** — Displays the OS Type of the hosts on this subnet in a pie chart.

Discovery Access reports
To run Discovery Access reports:

1. Click Discovery Access link in the Directly Discovered Data Index section of the Discovery Reports page.
2. Click the Reports drop down.
   The following Discovery Access-related reports are available:
   - **Discovery Access Summary** — Summary result state of the Endpoints.
   - **Discovery Access Detail** — Detailed end state of the Endpoints.

Individual Discovery Access reports
To run reports on individual Discovery Accesses in your system:

1. Click Discovery Access link in the Directly Discovered Data Index section of the Discovery Reports page. The Discovery Access List page is displayed.
2. Select a Discovery Access from the list.
3. Click the Reports drop down.
   The following Discovery Access-related reports are available:
   - **Discovery Method Timings** — Show timing per discovery method (does NOT include session establishment time).

Individual network interface reports
To run reports on individual network interfaces in your system:

1. Click the Hosts link in the Infrastructure Summary section of the Infrastructure page.
2. Select a Host from the list.
3. Select a Network Interface from the Hardware and Network section.
4. Click the Reports drop down.
   The following Discovery Access-related reports are available:
• Other Hosts on same Subnet — Shows other hosts that are on this subnet.
• Other Devices on same Subnet — Shows other interfaces that are on this subnet.
• Ports on Interface — Shows ports directly bound to this interface from last scan.
• Historical Ports on Interface — Shows ports directly bound to this interface from all available scans.
• Network Connections on Interface — Shows network connections directly bound to this interface from last scan.
• Historical Network Connections on Interface — Shows network connections directly bound to this interface from all available scans.

Individual load balancer instance reports
To run reports on individual load balancer instances in your system:

1. Click the Load Balancer Instances link in the Infrastructure Summary section of the Infrastructure page.
2. Select a Load Balancer Instance from the list.
3. Click the Reports drop down.
4. The following load balancing-related reports are available:
   • Load Balanced Hosts — Shows load balanced hosts.
   • Load Balanced Software — Shows load balanced Software Components and Software Instance.
   • Load Balancer Members not linked to a Host — Shows Load Balancer Members that are not linked to a Host.
   • Load Balancer Members not linked software — Shows Load Balancer Members that are not linked to a Software Component or Software Instance.

Individual storage system reports
To run reports on individual storage systems in your system:

1. Click the Storage System Instances link in the Infrastructure Summary section of the Infrastructure page.
2. Select a Storage System from the list.
3. Click the Reports drop down.
4. The following storage system-related reports are available:
   • Consumer Hosts

Using the Search and Reporting service
The searching and reporting service provides a high-level mechanism for retrieving data stored in the datastore. It is mainly driven by a query language that specifies which nodes to retrieve, and the attributes of those nodes to display.
This section describes how to enter search queries, and provides reference to the query language itself. The material is intended for software developers and users who are familiar with the data model and the taxonomy as well as query languages such as SQL.

- Using the Search service
- Using Query Language
- Context-sensitive reporting and linking

**Using the Search service**

This topic provides instructions for using the search service.

- To enter a search query
- To view current searches
- To cancel a search

**To enter a search query**

Search queries are entered by using the Enter Generic Query page.

1. Click the **Search** icon to the left of the Search box at the top right of the UI. The Search Options in the drop down panel are displayed.
2. Click the **Generic Search Query** link.
3. Enter the query in the text entry field.
4. Click **Run Query**.
   - The results are displayed in a report.
     - To export the results in a comma separated values (CSV) file, click **Export (CSV)**.
     - To export the results in an XML file, click **Export (XML)**.
     - In each case, a download dialog is displayed.
5. Save the file to your local file system.

If you have entered queries before, a list of the your recent queries is provided below the text entry box.

- Click on a link to fill in the text entry field so the query can be edited or run.
- If a previous query link is bookmarked (by right-clicking or dragging to a bookmark bar in your browser), it acts as a **bookmarklet**. Selecting the bookmark while viewing any BMC Atrium Discovery UI page runs the query on that BMC Atrium Discovery instance. You can bookmark a query on one BMC Atrium Discovery instance and run it on another one.

> **Note**

To access this page you require the ui/report/admin permission.
To view current searches

You can view all searches and reports that are currently in progress on the appliance. To do this, select **Search Management** from the Model section of the **Administration** tab.

The Search Management page shows the searches and reports that are currently in progress on the appliance, depending on the permissions that you have been assigned. A system user can see searches belonging to all users. A normal user can see just their own searches. This behavior is controlled by using the model/search/list and model/search/cancel permissions. See **Search Permissions** for more information.

The page shows searches from all sources, including those submitted through the Generic Query page, ones due to reports, and ones that are implicitly triggered while browsing in the UI. System users can also see searches performed by patterns, which are considered to belong to the **Reasoning Subsystem user**.

To cancel a search

The Search Management page lists any searches that you are permitted to view. Any search that you can cancel is shown with a **Cancel** button at the right hand side of its row. To cancel a search, click the **Cancel** button that corresponds to the search that you want to cancel.

When the search has been cancelled, the window in which the search was entered is refreshed with a Search Cancelled message.

You can also cancel a search that you have just started by navigating away from the search-in-progress spinner.

Attribute name conventions

In BMC Atrium Discovery the naming convention for attributes is to use only the lowercase letters a to z with underscores (\_) to separate words. Attribute names must not start or end with a space. Attribute names in searches can only be specified using the same characters. If you use other characters, for example a hyphen (-), your search will cause an error.

Using Query Language

This topic provides information and instructions for using the query language.

- Search expressions
- Example of a query
- LOOKUP expressions
- Metadata LOOKUP expressions
- Comments
- Literal strings
- Keywords
- Kind selection
Search expressions

The query language provides a natural mechanism for searching and processing the data model. The basic format of a search expression is:

```
SEARCH [in partition] <kinds> [where clause] [traversals] [ordering] [show clause] [processing]
```

- **[in partition]**: Used to search in a named partition.
- **<kinds>**: Used to specify the kinds of nodes (objects) to find.
- **[where clause]**: Used to filter the current set of nodes.
- **[traversals]**: Used to define a traverse from one node to another in order to access attributes and relationships from related nodes.
- **[ordering]**: Used to define a sort order for the results.
- **[show clause]**: Used to define the information to return in the search results.
- **[processing]**: Used to post-process the results of a search.

Example of a query

As a simple example, the following query retrieves all Host objects where the OS is Microsoft Windows. It displays each host's name and how much RAM it has; the results are sorted by name.

```
SEARCH Host WHERE os_type = 'Windows' ORDER BY name SHOW name, ram
```

The following example finds all nodes that mention Microsoft:

```
SEARCH * WHERE * HAS SUBWORD "microsoft"
```

In this example, both the ordering and show clauses are absent. The search service therefore uses the taxonomy definitions to choose defaults. The results are ordered by the label attributes of each node kind found and the attributes to show are set according to the corresponding summary lists. (If no definitions are given in the taxonomy for a node kind, the results are not sorted and just the node ids are shown.)

The following example searches the _System partition for Users:

```
search in "_System" FoundationUser show username
```
When writing search queries, you should be aware that an unconstrained search can have a serious performance impact on the appliance. For example, `SEARCH *` would return details of every node in the entire datastore!

The sets resulting from searches (and traversals) can be named and combined using set operations. This is described in Results Post Processing.

**LOOKUP expressions**

Instead of performing a `SEARCH`, a search can be performed with a `LOOKUP` that simply finds one or more nodes with their node id:

An example finding a single node:

`LOOKUP "845981da735bb875d1246b496e486f7374" SHOW hostname`

An example finding multiple nodes:

`LOOKUP "845981da735bb875d1246b496e486f7374", "88ee6097f60d047c3557dc4a6e486f7374", "88edfa88f60d047c3557dc4a6e486f7374" SHOW hostname`

A `LOOKUP` cannot have a `WHERE` clause. It is usually used in conjunction with one or more traversals.

**Metadata LOOKUP expressions**

`LOOKUP` is also used to retrieve a number of system metadata results:

<table>
<thead>
<tr>
<th><code>LOOKUP</code> version</th>
<th>Returns a single row of data with three columns: version number, release number, release code name</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>LOOKUP</code> partitions</td>
<td>Returns a single column of data in which the rows list the names of the partitions within the data store</td>
</tr>
<tr>
<td><code>LOOKUP</code> counts</td>
<td>Returns counts of nodes and relationships separated by node kind and partition</td>
</tr>
</tbody>
</table>

**Comments**

Search queries can contain comments on lines starting `//`. Everything is ignored from `//` to the end of line.

**Literal strings**

Literal strings used in search expressions can take a number of forms.

<table>
<thead>
<tr>
<th><code>string</code></th>
<th>A string terminated by an unescaped <code>\</code> character. Cannot include newlines.</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;string&quot;</td>
<td>A string terminated by an unescaped <code>&quot;</code> character. Cannot include newlines.</td>
</tr>
<tr>
<td>'''string'''</td>
<td>A string terminated by an unescaped <code>'''</code> character sequence. Can include newlines.</td>
</tr>
<tr>
<td>&quot;&quot;&quot;string&quot;&quot;&quot;&quot;</td>
<td>A string terminated by an unescaped <code>&quot;&quot;&quot;</code> character sequence. Can include newlines.</td>
</tr>
</tbody>
</table>
In normal string literals, escape characters start with backslash \ characters. Usual C-style escapes are permitted.
Strings can be 'qualified' to change their interpretation, by prefixing the string literal with a word as follows:

<table>
<thead>
<tr>
<th>raw</th>
<th>Backslash characters do not resolve to escape sequences.</th>
</tr>
</thead>
<tbody>
<tr>
<td>regex</td>
<td>Backslash characters do not resolve to escape sequences. Intended for use in MATCHES expressions.</td>
</tr>
<tr>
<td>path</td>
<td>Backslash characters do not resolve to escape sequences. Intended for use with filesystem paths.</td>
</tr>
<tr>
<td>unix_cmd</td>
<td>Expanded into a regular expression suitable for matching a UNIX command by prefixing with \b and suffixing with $.</td>
</tr>
<tr>
<td>windows_cmd</td>
<td>Expanded into a regular expression suitable for matching a Windows command by prefixing with (?i)\b and suffixing with .*exe$.</td>
</tr>
</tbody>
</table>

**Keywords**

In this document, query language keywords appear in upper case to make them stand out. Keywords are actually case-insensitive, so they can be specified in lowercase or mixed case.

**Note**

All other parts of query expressions are case sensitive.

To use an identifier that clashes with a keyword, prefix it with a $ character to prevent the parser reporting a syntax error:

SEARCH * WHERE $search = 123

For backwards-compatibility, keywords can also be escaped with the ~ character.

The keywords are as follows:

| AND    | Logical operator when defining conditions. See Logical Operators. |
| AS     | Modifier defining default heading shown. See The SHOW Clause. Use in function result naming. See Name binding. |
| BY     | Used in ORDER BY clause. See Ordering. |
| DEFINED | Used in Definition Boolean condition. See Conditions. |
| DESC   | Changes sort order from ascending to descending. See Ordering. |
| EXPAND | Used for repeated traversals. See Traversals. |
| EXPLODE | Used to 'explode' the items in the list into multiple output rows. See Explode. |
| FLAGS  | Used to modify behavior with destroyed nodes, result segmentation and other characteristics. See Search Flags. |
| HAS    | Used in substring and subword Boolean conditions. See Conditions. |
| **IN** | Used in containment Boolean conditions. See Conditions. Used with STEP when performing traversals, to move from a set of nodes to a set of relationships. See Traversals. |
| **IS** | Used in definition Boolean conditions. See Conditions. |
| **LIKE** | Deprecated synonym for MATCHES. |
| **LOCAL** | Used to show localized column headings in queries. See The SHOW Clause. |
| **LOOKUP** | Finds a single node. |
| **MATCHES** | Used in Boolean conditions when matching regular expressions. See Conditions. |
| **NODECOUNT** | Returns number of nodes when performing traverses. See NODECOUNT Expressions. |
| **NODES** | Returns a list of the traversed-to nodes. See NODECOUNT Expressions. |
| **NOT** | Used in substring and subword Boolean conditions. See Conditions. Logical operator when defining conditions. See Logical Operators. |
| **OR** | Logical operator when defining conditions. See Logical Operators. |
| **ORDER** | Used in ORDER BY clause. See Ordering. |
| **OUT** | Used with STEP when performing traversals, to move from a set of relationships to a set of nodes. See Traversals. |
| **PROCESS** | Used to summarize or modify the search results. See Results after processing. |
| **PROCESSWITH** | Used to summarize or modify the search results. See Results after processing. |
| **SEARCH** | Runs a search. See Using Query Language. |
| **SHOW** | Defines the columns to return in the search results. See The SHOW Clause. |
| **STEP** | Used with IN and OUT when performing traversals, to move between nodes and relationships. See Traversals. |
| **SUBSTRING** | Used in substring Boolean condition. See Conditions. |
| **SUBWORD** | Used in subword Boolean condition. See Conditions. |
| **SUMMARY** | Used to show the taxonomy-defined summary list in the search results. See The SHOW Clause. |
| **TAXONOMY** | Used to order by the taxonomy-defined label and to refer to named attribute lists. See Ordering. |
| **TRAVERSE** | Used to traverse from one node to another in order to access attributes and relationships from related nodes. See Traversals. |
| **WHERE** | Filters the current set of nodes according to a Boolean condition. See Logical and arithmetic expressions. |
| **WITH** | Used with the PROCESS keyword to specify the post processing function to use. See Results after processing. Used in function result naming. See Name binding. |

**Kind selection**

A search expression must specify the kinds of nodes to search. The specification must either be a single * character, meaning to search all kinds, or a comma-separated list of node kinds. The majority of queries specify a single kind.
**Key expressions**

The *WHERE*, *ORDER BY* and *SHOW* clauses often use simple attribute names, as in the examples in the sections above. In some cases, however, a query needs more complex specifications.

The first option is to use a 'key expression' to retrieve information from related nodes. Key expressions start with a # character.

The key expression formats are:

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#role:rel:role:kind.attr</td>
<td>Traverses from the node via the roles and relationships, returning the attribute of the target node. If more than one target can be reached, returns a list of up to 10 (or an alternative limit chosen in the system options).</td>
</tr>
<tr>
<td>#role:rel:role:kind</td>
<td>As above, but returns a 'NodeHandle' for the target node, rather than an attribute of it, useful as a function argument.</td>
</tr>
<tr>
<td>#role:rel.attr</td>
<td>Step in to a relationship and get its attribute.</td>
</tr>
<tr>
<td>#role:rel</td>
<td>Step in to a relationship and return a 'RelationshipHandle' for it, to pass into functions.</td>
</tr>
<tr>
<td>#:role:kind.attr</td>
<td>Step out of a relationship and get the destination node's attribute.</td>
</tr>
<tr>
<td>#:role:kind</td>
<td>Step out of a relationship and return the destination 'NodeHandle'.</td>
</tr>
<tr>
<td>#id</td>
<td>Return the node's id in hex string format.</td>
</tr>
<tr>
<td>##</td>
<td>Returns the node ID in binary format.</td>
</tr>
<tr>
<td>#</td>
<td>Returns a 'NodeHandle' for the node itself, to pass into various functions.</td>
</tr>
<tr>
<td>#node</td>
<td>Returns the node itself.</td>
</tr>
<tr>
<td>#&quot;&lt;fmt&gt;&quot;(a,b,c)</td>
<td>Applies Python format string to the attributes (which can be key expressions) in the list.</td>
</tr>
</tbody>
</table>

So, extending the example of finding Windows hosts, this expression shows the IP addresses of the hosts' network interfaces:

```
SEARCH Host WHERE os_type = 'Windows' ORDER BY name SHOW name, ram, #DeviceWithInterface:DeviceInterface:InterfaceOfDevice:NetworkInterface.ip_addr
```

The traversal syntax permits components to be wild-carded by leaving them blank. For example, the following query shows the names of all software running on a Host, both Business Application Instances and Software Instances:

```
SEARCH Host SHOW name, #Host:HostedSoftware:RunningSoftware:.name
```
Key expression name binding

Repeated key expressions can lead to long impenetrable search expressions. To avoid this, they can be bound to names. For example:

```
SEARCH Host
WITH #Host:HostedSoftware:RunningSoftware:SoftwareInstance AS si
SHOW name, EXPLODE si.type, si.version, si.count
```

Query Language Functions

Another option for manipulating attributes is to apply functions to them. Functions are applied using the familiar parenthesis syntax:

```
SEARCH Host SHOW len(name)
```

Value manipulation

The following functions operate on normal attribute values.

abs(value)

Returns the absolute value of an integer.

bin(value, bins [, legend])

Separates numeric values into 'bins', based on a list of values providing the bin boundaries, for example, a search like this:

```
SEARCH Host SHOW name, ram, bin(ram, [64, 512, 1024, 4096])
```

gives results like

<table>
<thead>
<tr>
<th>Host Name</th>
<th>RAM</th>
<th>Bin Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>rs6000</td>
<td>1888</td>
<td>1024-4096</td>
</tr>
<tr>
<td>jpl64app</td>
<td>896</td>
<td>512-1024</td>
</tr>
<tr>
<td>sol10x86</td>
<td>288</td>
<td>64-512</td>
</tr>
<tr>
<td>linux-mandr-9-1</td>
<td>32</td>
<td>Less than 64</td>
</tr>
<tr>
<td>lonvmserv02</td>
<td>12288</td>
<td>4096 and more</td>
</tr>
</tbody>
</table>

The optional `legend` parameter allows the bins to be named. The list must have one more item than the `bins` list. For example:

```
SEARCH Host SHOW name, ram, bin(ram,
```

boolToString(value)
Interprets its argument as a Boolean and returns "Yes" or "No".

booleanLabel(value, true_label, false_label, other_label)
A more advanced version of boolToString, which lets you choose which label to use for the True, False, and None cases.

duration(start_time, end_time)
Calculates the amount of time elapsed between the two dates and returns the result as a string of the form \textit{days.hours.minutes.seconds}.

currentTime()
Returns the number of 100 nanosecond intervals since 15 October 1582. This example query returns the hosts created in the last 7 days:

\begin{verbatim}
search Host where creationTime(#) > (currentTime() - 7*24*3600*10000000) show name, time(creationTime(#) as CreateTime
\end{verbatim}

defaultNumber(value, default_value)
Attempts to convert the value into a number and if it is not possible returns the default value.

extract(value, pattern [, substitution ])
Performs a regular expression extraction. The value is matched against the pattern. If no substitution is given, returns the part of the value that matched the pattern. If substitution is given, it specifies a string including group references of the form \textbackslash 1, \textbackslash 2, and so on, that are filled in with the corresponding groups in the pattern. If the value does not match the pattern, returns an empty string. Strings containing backslashes must be specified with a regex qualifier, for example, regex "\textbackslash 1".
**fmt(format, ...)**

The first argument is a Python format string, the remaining arguments are values to interpolate into it. The result is equivalent to a key expression of the form `"format"(...)`, except that the arguments to the `fmt()` function can be the results of other functions. That is, these two expressions are equivalent:

```
SEARCH Host SHOW "%s: %s"(name,ram)
SEARCH Host SHOW fmt("%s: %s",name,ram)
```

Whereas the following expression can only be done with `fmt()` because it calls the `len()` function:

```
SEARCH Host SHOW fmt("%s: %d",name,len(name))
```

The following example shows the `fmt()` function used to show the results of floating point arithmetic:

```
fmt( ".4f", float(size)/1048576) as 'Size GB'
```

**fmt_map(format, items)**

Applies a format string to all items in a list. The `format` argument must be a python format string with only one variable. `items` is the list of items to interpolate into the string. Returns a list of strings as a result.

**formatNumber(value, singular, plural, other)**

If `value` is 1, returns `singular`, if `value` is a number other than one, return `plural` with the value inserted at the mandatory `%d` inside it, and if `value` is not a number return `other`. e.g.

```
SEARCH Host SHOW name, patch_count, formatNumber(patch_count, "One patch", "%d patches", "Unknown patches")
```
formatQuantity(value, format)

Takes a value and applies user friendly formatting, putting the values into bits, bytes, KiB, Mb, and so on. The value, an `int` or a `float` is formatted according to a number of fixed `format` parameters. Format parameters are all literal strings. If non-numeric values such as strings or `None` are passed then they are returned unmodified.

The format parameters and examples are shown in the following table:

<table>
<thead>
<tr>
<th>Value</th>
<th>Format parameter</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>&quot;1000&quot;</td>
<td>1 k</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;1000&quot;</td>
<td>1 M</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;1024&quot;</td>
<td>1000</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;1024&quot;</td>
<td>976.6 Ki</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;B1000&quot;</td>
<td>1 kB</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;B1000&quot;</td>
<td>1 MB</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;B1024&quot;</td>
<td>1000 bytes</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;B1024&quot;</td>
<td>976.6 KiB</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;b1000&quot;</td>
<td>1 kb</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;b1000&quot;</td>
<td>1 Mb</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;b1024&quot;</td>
<td>1000 bits</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;b1024&quot;</td>
<td>976.6 Kib</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;bitrate&quot;</td>
<td>1 kbit/s</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;bitrate&quot;</td>
<td>1 Mbit/s</td>
</tr>
<tr>
<td>1000</td>
<td>&quot;byterate&quot;</td>
<td>1 kB/s</td>
</tr>
<tr>
<td>1000000</td>
<td>&quot;byterate&quot;</td>
<td>1 MB/s</td>
</tr>
</tbody>
</table>

The following example shows applying friendly formatting the raw capacity of a StoragePool:

```sql
SEARCH StoragePool SHOW name, total_raw_capacity, formatQuantity( total_raw_capacity, "10 as 'Size 1000', formatQuantity( total_raw_capacity, "B1024") as 'Size B1024'
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Total Raw Capacity</th>
<th>Size 1000</th>
<th>Size B1024</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pool 1</td>
<td>1435374714880</td>
<td>1.4 T</td>
<td>1.3 TiB</td>
</tr>
<tr>
<td>Pool 2</td>
<td>27686186726640</td>
<td>27.7 T</td>
<td>25.2 TiB</td>
</tr>
<tr>
<td>Pool 3</td>
<td>384832081920</td>
<td>384.8 G</td>
<td>358.4 GiB</td>
</tr>
<tr>
<td>Name</td>
<td>Total Raw Capacity</td>
<td>Size 1000</td>
<td>Size B1024</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------</td>
<td>-----------</td>
<td>------------</td>
</tr>
<tr>
<td>Pool 4</td>
<td>0</td>
<td>0</td>
<td>0 bytes</td>
</tr>
</tbody>
</table>

**formatTime(time_value, format)**

Converts the internal time format to a string, based on the `format` specification, and converting into the appliance's time zone. The format is specified using Python's `strftime` format. For example, a search like this:

```python
SEARCH Host SHOW name, formatTime(last_update_success, "%d %B %Y")
```

Gives results:

<table>
<thead>
<tr>
<th>Hostname</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>lonvmserv03</td>
<td>15 January 2009</td>
</tr>
<tr>
<td>rs6000</td>
<td>12 January 2009</td>
</tr>
<tr>
<td>sol10x86</td>
<td>13 January 2009</td>
</tr>
</tbody>
</table>

**formatUTCTime(time_value, format)**

Identical to `formatTime`, except that it does not perform timezone conversion.

**friendlyTime(time_value)**

Converts the internal time format into a human readable string, taking into account time zones and daylight saving times, based on the time zone of the appliance.

**friendlyUTCTime(time_value)**

Converts the internal time format into a human readable string, without converting the time to account for time zones and daylight saving times.

**friendly_duration(duration)**

Takes a duration (that is, one time minus another) and returns a human readable string of the result, such as '3 days' or '1 month' or '30 seconds'. The result is not intended to be precise, but to be quickly understood by a person.

**get(item, attribute [, default ])**

Retrieve `attribute` from `item`. If the item does not have the specified attribute, returns `default` if it is specified, or None if not.

**hash(value)**

Returns the MD5 hash of the specified value.
int(value [, default])
Converts a string form of an integer to an integer. Works on lists. Optionally supports a second argument, which if present will be used if the string cannot be converted.

join(value, separator)
Build a string out of a list by concatenating all the list elements with the provided separator between them.

leftStrip(value)
Returns the value with white space stripped from the start.

len(value)
Returns the length of a string or list.

lower(string_value)
Returns a lower-case version of a string.

parseTime(time_string)
Converts a date/time string into the internal format, without time zone conversion.

parseUTCTime(time_string)
Converts a date/time string into the internal format. Identical to parseTime.

parseLocalTime(time_string)
Converts a date/time string into the internal format, taking into account time zones and daylight saving times, based on the time zone of the appliance.

replace(value, old, new)
Modifies value, replacing all non-overlapping instances of the string old with new.

recurrenceDescription(recurrence)
Converts a recurrence object to a human readable string.

rightStrip(value)
Returns the value with white space stripped from the end.

single(value)
If value is a list, return just the first item of it; otherwise return the value unchanged. This is useful when following key expressions that usually return a single item, but occasionally return multiple. e. g.

```plaintext
search Host show name, single(#InferredElement:Inference:Primary:HostInfo.uptimeSeconds)
```
size(value)
Returns the size of a list or string. A synonym for `len()`.

sorted(values)
Returns the sorted form of the given list.

split(value [, separator ])
Split a string into a list of strings. Splits on white space by default, or uses the specified `separator`.

str(value)
Converts its argument to a string.

strip(value)
Removes white space at the start and end of the value.

sumValues(values)
Sums a list of values. For example, to total the `count` attributes of the Software Instances related to each Host:

```
search Host show name, sumValues(#Host:HostedSoftware:RunningSoftware:SoftwareInstance.count)
```

time(time_val)
Marks a number to indicate that it is a time. The values returned by functions such as `currentTime` and `parseTime` are large numbers (representing the number of 100 nanosecond intervals since 15 October 1582), which can be manipulated by other functions and compared to each other. To return them in results in a way that the UI knows that they are times, they must be annotated as times using the `time` function.

toNumber(value [, base ])
Converts a string into a number. If `base` is given, uses the specified base for the conversion, instead of the default base 10.

toText(value [, base [, width ] ])
Converts a number to a string. If `base` is given, the conversion uses the specified base. Only bases 8, 10 and 16 are valid. If `width` is given, the string is padded so that it contains at least `width` characters, padding with spaces on the left.

unique(values)
Returns a list containing the unique values from the provided list.
upper(string_value)

Returns an upper-case version of a string.

value(item)

Returns `item` unchanged. This is only useful to bind a non-function result to a name, as described in Name binding.

whenWasThat()

Converts the internal time format to something easily readable, like ‘1 hour ago’, ‘2 weeks ago’, and so on.

Node manipulation

These functions must be passed nodes with key expressions, often just a single `#` to represent the current node:

```
SEARCH Host SHOW name, keys(#)
```

```
// Use the time() function to tell the UI that the result is a time.
SEARCH Host SHOW name, time(modified(#))
```

destroyed(node)

Returns True if the node has been destroyed, False if not. Returns [invalid node] if the argument is not a node. Works on lists of nodes as well, returning a list of boolean values. (See the section on Search Flags that permit searching destroyed nodes.)

hasRelationship(node, spec)

Takes a node and a traversal specification. Returns True if the node has at least one relationship matching the specification; False if not. Works on lists of nodes as well.

id(node)

DEPRECATED function to return a node id in string form. Use `#id` to return a node's id.

keys(node)

Returns a list of the keys set on the node. Returns [invalid node] if the argument is not a node. Works on lists of nodes as well, returning a list of lists of keys.

kind(node)

Returns the kind of the node. Returns [invalid node] if the argument is not a node. Works on lists of nodes as well, returning a list of kinds.
label(node)
Returns the node's label, as defined in the taxonomy. Works on lists of nodes as well, returning a list of labels.

modified(node)
Returns the node's last modified time in the internal numeric format, this includes *any* modification, including relationships to the node. The modified() function works on lists of nodes as well, returning a list of times.

⚠️ Modification times and host nodes
At the end of each discovery run, the automatic grouping feature considers all the Hosts, and builds a new set of automatic grouping relationships. It commits one big transaction that adjusts all the relationships to all Hosts, so every Host node usually has the same modification time.

provenance(node, attribute [, show_attribute])
Follows provenance relationships from the node, finding the evidence node that provided the specified attribute. If the `show_attribute` is given, returns the specified attribute of the evidence node; if not, returns a handle to the node itself.

NODECOUNT and NODES
In addition to the functions above, the NODECOUNT and NODES keywords, defined in Traversals, behave like functions in some respects.

History functions
The following history-related functions are currently available.

creationTime(node)
Returns the number of 100 nanosecond intervals between 15 October 1582 and the time the node was created. Also works on lists of nodes.

createdDuring(node, start, end)
Returns true if the node was created during the time range specified with start and end. For example, to find all the application instances created between 1st July and 10th July 2008.

```sql
SEARCH BusinessApplicationInstance
WHERE createdDuring(#, parseTime("2008-07-01"), parseTime("2008-07-10"))
```
destroyedDuring(node, start, end)
Returns true if the node was destroyed during the time range specified with start and end. To find all the application instances destroyed between 1st July and 10th July 2008:

```sql
SEARCH FLAGS (include_destroyed, exclude_current) BusinessApplicationInstance
WHERE destroyedDuring(#, parseTime("2008-07-01"), parseTime("2008-07-10"))
```

destructionTime(node)
Returns the time the node was destroyed in the internal time format. If the node is not destroyed, returns 0. Works on lists of nodes as well, returning a list of destruction times.

eventOccurred(node, start, end)
Takes a node and two times in the internal format. Returns True if the node was modified between the specified times; False if not. Works on lists of nodes as well. Returns [invalid time] if the times are invalid.

Specialist history functions
The following history functions can be used for specialist purposes:

newInAttr(node, attr, timeA, timeB)
Retrieves the node's specified attribute at the two times. The attribute is expected to contain a list. Returns a list containing all items that were present in the list at timeB that were not present at timeA.

attrSpread(node, attr, timeA, timeB)
Returns a list containing all unique values that the attribute has had between the two times.

newInAttrSpread(node, attr, timeA, timeB, timeC)
A cross between newInAttr and attrSpread. Returns a list of values for the attribute that existed at any time between timeB and timeC, but which did not exist at any time between timeA and timeB.

historySubset(nh, timeA, timeB, attrs, rels)
Reports on a subset of the node history between the two times. attrs is a list of attribute names to report; rels is a list of colon-separated relationship specifications to report. For example, the following query will show changes to os_type, os_version, and hosted SoftwareInstances for a collections of Hosts:

```sql
SEARCH Host SHOW
  name,
  historySubset(#, 0, currentTime(),
    ["os_version", "os_type"],
    ["Host:HostedSoftware:RunningSoftware:SoftwareInstance"])
```
See also the post-processing function `displayHistory` in Results after processing.

System interaction

These functions allow access to other aspects of the BMC Atrium Discovery system.

`fullFoundationName(username)`
Returns the full name of the user with the given user name or None if no such user exists.

`getOption(key)`
Returns the value of the system option `key`.

Link functions

When search results are shown in the UI, each cell in the result table is usually a link to the node corresponding to the result row. These functions allow other links to be specified:

`nodeLink(link, value)`

`link` is a node id or node reference, for example the result of a key expression; `value` is the value to display in the UI and to be used in exports. For example, to create a table listing Software Instances and their Hosts, with links from the host names to the Host nodes:

```
SEARCH SoftwareInstance SHOW
    name AS "Software Instance",
    nodeLink(#RunningSoftware:HostedSoftware:Host:Host, #RunningSoftware:HostedSoftware:Host:Host.name) AS "Host"
```

`queryLink(link, value)`

`link` is a search query to execute; `value` is the value to display in the UI and to be used in exports.

Logical and arithmetic expressions

Logical and arithmetic expressions can be used in \texttt{SHOW} clauses and \texttt{WHERE} clauses.

Truth

In \texttt{WHERE} clauses, the following values are considered False:

| Boolean False | Number zero | Empty string | Empty list | None (for example, missing attribute) |
All other values are considered True. This allows simple \texttt{WHERE} clauses that choose only useful values. For example, to find all Software Instance nodes with a populated version:

\begin{verbatim}
SEARCH SoftwareInstance WHERE version SHOW type, version
\end{verbatim}

\section*{Expressions}

The following logical and arithmetic expressions are supported:

\begin{center}
\begin{tabular}{|l|l|}
\hline
\textbf{Equality}   & \texttt{a = b} \\
\hline
\textbf{Inequality} & \texttt{a <> b} \\
\hline
\textbf{Comparison} & \texttt{a > b} \\
& \texttt{a >= b} \\
& \texttt{a < b} \\
& \texttt{a <= b} \\
\hline
\textbf{Arithmetic} & \texttt{a + b} \\
& \texttt{a - b} \\
& \texttt{a * b} \\
& \texttt{a / b} \\
\hline
\textbf{Subwords}   & \texttt{a HAS SUBWORD b} \\
\hline
\end{tabular}
\end{center}

Case-insensitive subword or phrase test. \texttt{a} is split into a word list using non-alphanumeric characters to identify the word boundaries.

For a single word search, the condition is true if \texttt{b} is in the list of split words.

For a multi-word search:

- To find a number of words in the text, disregarding the word order, enclose the words with quotation marks, for example:

\begin{verbatim}
os_type has subword "Red Hat Linux"
\end{verbatim}

This condition is true if all the words "Red", "Hat", and "Linux" are found as subwords of the full text, not limiting results to the exact phrase match. For example, if \texttt{os_type} is "Red Hat Enterprise Linux", the search in the example above returns true.

- To find an exact phrase in the text, taking into account the word order, enclose the phrase in square brackets, like in the following example:

\begin{verbatim}
os_type has subword ["Red Hat Linux"]
\end{verbatim}

This condition is true if the exact phrase is found as a subphrase of the full text, matching the word order. For example, if \texttt{os_type} is "Red Hat Enterprise Linux", the condition in the example above returns false.

When used in a \texttt{WHERE} clause, \texttt{HAS SUBWORD} uses the full text indexes maintained by the datastore. It is one of the most efficient ways to find nodes.
b HAS SUBSTRING a

Case-insensitive substring test. Where possible HAS SUBSTRING uses the full text indexes maintained in the data store, but it is always less efficient than HAS SUBWORD.

<table>
<thead>
<tr>
<th>Regular expression match</th>
<th>a MATCHES b</th>
</tr>
</thead>
<tbody>
<tr>
<td>a LIKE b</td>
<td></td>
</tr>
</tbody>
</table>

The condition is True if a matches regular expression b. LIKE is a deprecated synonym of MATCHES. MATCHES cannot often use the datastore indexes, and is therefore by far the slowest mechanism for finding nodes. If at all possible queries should use HAS SUBWORD instead of MATCHES.

<table>
<thead>
<tr>
<th>Containment</th>
<th>a IN b</th>
</tr>
</thead>
<tbody>
<tr>
<td>a NOT IN b</td>
<td></td>
</tr>
</tbody>
</table>

The condition is True if a is/is not in b, where b is a list.

<table>
<thead>
<tr>
<th>Containment</th>
<th>a IN [b, c, d...]</th>
</tr>
</thead>
<tbody>
<tr>
<td>a NOT IN [b, c, d...]</td>
<td></td>
</tr>
</tbody>
</table>

The condition is True if a is/is not in the specified list.

<table>
<thead>
<tr>
<th>Definition</th>
<th>a IS DEFINED</th>
</tr>
</thead>
<tbody>
<tr>
<td>a IS NOT DEFINED</td>
<td></td>
</tr>
</tbody>
</table>

The condition is True if the node has an attribute a (with any value).

<table>
<thead>
<tr>
<th>Inverse</th>
<th>NOT a</th>
</tr>
</thead>
<tbody>
<tr>
<td>True if a is considered False; False if it is considered True.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>And</th>
<th>a AND b</th>
</tr>
</thead>
<tbody>
<tr>
<td>Considered True if both a and b are considered True. If a is considered True, the value of the expression is b; if a is considered False, the value of the expression is a.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Or</th>
<th>a OR b</th>
</tr>
</thead>
<tbody>
<tr>
<td>Considered True if either a is considered True or b is considered True. If a is considered True, the value of the expression is a; otherwise the value of the expression is b.</td>
<td></td>
</tr>
</tbody>
</table>

### Precedence

Operator precedence works as you would expect. Parentheses ( and ) are used to explicitly group operations. Otherwise, in arithmetic expressions, * and / take precedence over + and -. In logical expressions, OR has lowest precedence, followed by AND, followed by NOT, followed by all the other operators. i.e. this expression

\[
\text{NOT } a > b \text{ OR } c \text{ HAS SUBWORD } d \text{ AND } e = 10
\]

is equivalent to

\[
\text{((NOT } (a > b)) \text{ OR } ((c \text{ HAS SUBWORD } d) \text{ AND } (e = 10)))
\]

### Logical expressions in SHOW clauses

Logical expressions can be used in SHOW clauses. To do so, they must be enclosed in parentheses, e.g.
The behavior of **AND** and **OR** in returning one of their input parameters can be useful in handling missing attributes, to avoid output of "None" when an attribute is not available, for example:

```plaintext
SEARCH Host SHOW name, (ram > 1024)
```

Logical expression name binding

Logical expressions can be bound to names. For example:

```plaintext
SEARCH Host
WITH (virtual AND "yes" OR "no") AS is_virtual
SHOW name, @is_virtual
```

Name binding

It is sometimes necessary to use the result of calling a function more than once, for example using a result as part of a **WHERE** clause, and then using the value in a **SHOW** clause. To avoid repeated function evaluation, the result of a function can be bound to a name using a **WITH** clause. The result can then be referred to later by name, prefixed with an @ character. For example:

```plaintext
SEARCH Host
WITH creationTime(#) AS ctime
  WHERE @ctime > parseTime("2007-06-01")
SHOW name, os, @ctime
```

Only function results can be bound to names in this way. It is not possible to directly use a logical or arithmetic expression. To do so, you can use the `value()` function as a wrapper around the expression:

```plaintext
SEARCH Host
WITH value(currentTime() - creationTime(#)) AS offset
```
Traversals

A SEARCH expression creates a set of nodes, and a LOOKUP makes a set with just one node in it. With that set, a query can TRAVERSE from each of the nodes to a new set of related nodes. The format of a TRAVERSE expression is

\[
\text{TRAVERSE role:rel:role:kind [WHERE clause]}
\]

The WHERE clause allows the new set to be further filtered. For example, to find all software instances that are running on Linux hosts:

```
SEARCH Host WHERE os_type HAS SUBWORD "Linux"
TRAVERSE Host:HostedSoftware:RunningSoftware:SoftwareInstance
```

As with key expressions, elements of the traversal can be omitted as a wildcard mechanism. All matching relationships are followed, so this example finds software running on Linux hosts, and virtual machine software containing the hosts:

```
SEARCH Host WHERE os_type HAS SUBWORD "Linux"
TRAVERSE :::SoftwareInstance
```

Adding some filtering:

```
SEARCH Host WHERE os_type HAS SUBWORD "Linux"
TRAVERSE :HostedSoftware::SoftwareInstance
WHERE type HAS SUBSTRING "oracle"
```

The attributes that are accessed in the TRAVERSE WHERE clause are attributes of the nodes that have been traversed to, not attributes of the original nodes. The original nodes have been discarded from the set.

Traversals can be chained, so we can now find any business application instances related to the software instances:
SEARCH Host WHERE os_type HAS SUBWORD "Linux"
TRAVERSE :HostedSoftware::SoftwareInstance
WHERE name HAS SUBSTRING "oracle"
TRAVERSE ContainedSoftware:SoftwareContainment:
SoftwareContainer:BusinessApplicationInstance

At each stage, the result of a traversal is a set, so each node appears only once in the set, even if it is reached via relationships from more than one node.

A TRAVERSE clause performs one single traverse from each node in the current set. Sometimes it is useful to repeatedly traverse to find all the nodes that can be reached via particular relationships, using an EXPAND clause. This is useful for finding all applications that depend on a particular other application, for example:

SEARCH BusinessApplicationInstance WHERE <some condition>
EXPAND DependedUpon:Dependency:Dependant:BusinessApplicationInstance

The set now contains the original nodes, plus all the nodes that depend upon them, via any number of intermediate dependencies. The EXPAND keeps performing traversals until no more nodes are found. The EXPAND always keeps the original set. If you want to EXPAND but not keep the original set, use a TRAVERSE followed by an EXPAND:

SEARCH BusinessApplicationInstance WHERE <some condition>
TRAVERSE DependedUpon:Dependency:Dependant:BusinessApplicationInstance
EXPAND DependedUpon:Dependency:Dependant:BusinessApplicationInstance

Just like a TRAVERSE clause, an EXPAND can have a WHERE clause. The filtering of the WHERE clause happens once the EXPAND has completed, not on every traversal iteration (since if it did it every iteration, it might never complete).

TRAVERSE and EXPAND move from one set of nodes to another set of nodes. Sometimes, you want to stop off at a relationship along the way. This is achieved with STEP IN, which replaces the set of nodes with a set of relationships. Given a set of relationships, you can STEP OUT to another set of nodes. To find critical Dependency relationships:

SEARCH BusinessApplicationInstance WHERE <some condition>
STEP IN DependedUpon:Dependency WHERE critical
To find the BusinessApplicationInstances that have a critical dependency to the ones matched in the WHERE clause:

```
SEARCH BusinessApplicationInstance WHERE <some condition>
STEP IN DependedUpon:Dependency WHERE critical
STEP OUT Dependant:BusinessApplicationInstance
```

### NODECOUNT Expressions

Sometimes it is useful to filter, sort, or show the number of nodes related to each node in the set. This is achieved with a **NODECOUNT** expression. It behaves like a function, except that its arguments are a traversal clause. To show the number of dependencies of each Business Application Instance:

```
SEARCH BusinessApplicationInstance SHOW name,
NODECOUNT(TRAVERSE DependedUpon:Dependency:
    Dependant:BusinessApplicationInstance)
```

**NODECOUNT** accepts all the traversal expressions described above, including chaining and where clauses.

**NODECOUNT** is the correct way to filter based on properties of related nodes, rather than using a key expression in a **WHERE** clause. The following is a way to find all Linux Hosts running products from the Apache foundation, for example:

```
SEARCH Host
WHERE os_type HAS SUBWORD "linux" AND
NODECOUNT(TRAVERSE Host:HostedSoftware:RunningSoftware:SoftwareInstance
    WHERE type HAS SUBWORD "apache")
```

Similar to **NODECOUNT**, the **NODES** keyword performs a traversal and returns a list of the traversed-to nodes. It is only used for internal purposes since, in general, lists of nodes cannot be handled in the search service.

### Search Flags

Certain aspects of search behavior can be modified by setting 'flags' for the search. Flags are applied with the **flags** keyword, and affect the whole search:

```
SEARCH FLAGS (flag1, flag2, ...) Host WHERE ...
```
Flags can also be applied to traversals, in which case they apply to just the traversal:

```
SEARCH Host WHERE os_type HAS SUBWORD "linux"
TRAVERSAL FLAGS (include_destroyed) ::::DiscoveryAccess
```

The following flags are available:

**include_destroyed**

Normally, nodes and relationships that are marked as destroyed are excluded from searches and traversals. The **include_destroyed** flag means that destroyed nodes and relationships are included.

**exclude_current**

With **include_destroyed**, searches involve both current and destroyed nodes; with the additional **exclude_current** flag, current nodes are excluded. **exclude_current** only makes sense when used in conjunction with **include_destroyed**, otherwise everything is excluded.

**no_segment**

Normally, a search that finds multiple target nodes segments its results so that each node kind is in a separate result set. **no_segment** prevents the segmentation, meaning the results are in just one set. This is generally only useful when the node kinds are similar in some way, otherwise it is impossible to define a suitable **SHOW** clause for the combined results.

**find_relationships**

Searches normally find nodes with the specified kinds. The **find_relationships** flag causes searches to find relationships instead.

**suppress_default_links**

When presented in the UI, search results are normally presented so that each row is a link to the node from which the data came. The **suppress_default_links** flag removes these default links.

**Ordering**

After processing a **SEARCH** or **LOOKUP** and any traversals, the set of nodes is in an arbitrary order. If the query expression contains an **ORDER BY** clause, the set is sorted according to the clause. The clause is a comma separated list of attributes (or key expressions, function calls, and so on). The set is sorted according to the first attribute in the list; those that are identical by the first attribute are further sorted according to the second attribute in the list, and so on.

```
SEARCH Person ORDER BY surname, name SHOW #"%s %s"(name, surname)
```
Sorting is normally in ascending order; the order can be reversed with the DESC modifier:

```
SEARCH Person ORDER BY surname DESC, name SHOW #"%s %s"(name, surname)
```

If the search expression does not have an ORDER BY clause, it is returned in an arbitrary order unless the taxonomy is consulted for the SHOW clause (see below). In that case, the set is sorted by the taxonomy-defined label attribute.

**Named taxonomy attribute lists**

The taxonomy supports multiple named attribute lists. These can be referred to in queries using the TAXONOMY keyword. For example:

```
SEARCH Host SHOW TAXONOMY "report_medium_detail"
```

or

```
SEARCH Host SHOW TAXONOMY "report_high_detail"
```

**The SHOW Clause**

The SHOW clause contains a list of columns to return in the search results. The clause is a comma separated list of attribute names, key expressions, functions or other expressions. For some reports, it is useful to set headings for the columns. By default, the heading for a column is the name of the attribute in that column, or a simple string describing how the value was arrived at. The default heading can be overridden with the AS modifier:

```
SEARCH Host SHOW name AS "Host name", os AS "Operating system"
```

If the SHOW clause is absent, the columns are set according to the summary list as defined in the taxonomy. If the SHOW clause is a single * character, the columns are set to the field order defined in the taxonomy. In both of those cases, if there is no entry in the taxonomy for the node kinds, no columns are set, and results can not be retrieved (although the nodes themselves can still be accessed).
To show the taxonomy-defined summary list defined plus some other attributes, use the `SUMMARY` keyword in a `SHOW` clause. This query shows the OS of a Host, followed by the normal taxonomy summary:

```plaintext
SEARCH Host SHOW os, SUMMARY
```

If columns are not named with `AS`, the columns are named using information from the taxonomy, as long as the locale is known. Searches performed through the UI take the locale from the user’s profile information. In other cases, such as exports, there is no default locale. The locale can be specified in the query:

```plaintext
SEARCH Host LOCALE "en_GB" SHOW name, os
```

If the locale is not known, or is is specified as the empty string, the column headings are based on the attribute names in the `SHOW` clause.

**Explode**

Normally, if an attribute contains a list, or a key expression traversal that leads to multiple nodes, all items in the list appear in a single cell in the results. Sometimes, particularly when exporting data to other systems, it can be useful instead to ‘explode’ the items in the list into multiple output rows. The result is similar to the result of a join in a relational database. This query produces a table of network interfaces with one fully-qualified domain name per row:

```plaintext
SEARCH NetworkInterface SHOW ip_addr, EXPLODE fqdns
```

When exploding an entry, one row is created for each item in the exploded list; if the exploded list is empty, no rows are created.

A common requirement when exporting data for import into another system is to extract the node identifiers for all nodes related to a set of nodes, so the graph structure can be reconstructed. This can be achieved by exploding the related node identifiers. This query produces a table mapping Host node ids to all the Software Instances running on them:

```plaintext
SEARCH Host SHOW name, #id, EXPLODE #Host:HostedSoftware:RunningSoftware:SoftwareInstance.
```
Key expression traversals leading to multiple nodes can be limited in the system settings so that not all target nodes are used (although by default there is no limit). When exploded, key expressions have no traversal limit.

All key expressions with the same traversal specifications are handled as a group, meaning that multiple attributes can be selected, for example:

```
SEARCH Host
SHOW name,
    EXPLODE #:::SoftwareInstance.name,
    #:::SoftwareInstance.instance
```

The result has one row per software instance. Even though the second key expression is not explicitly exploded, it is implicitly exploded since it shares the traversal with the exploded attribute.

You should not use explode on more than one independent attribute. If you do this, all possible combinations of the attribute values are exploded. This can produce very large datasets and can impact performance considerably.

**Results after processing**

After a query has completed, resulting in one or more sets of nodes, the set can be post-processed to summarize the results or otherwise modify them. This is achieved using a `PROCESSWITH` clause. (For backwards compatibility, `PROCESSWITH` can be specified as two words `PROCESS` WITH.)

⚠️ **Optional parameters**

In the following example, some function parameters are shown with a value, such as `min=0`. This indicates that the parameters are optional, and the values are the default used if values are not provided. The key=value syntax is not part of the search syntax. Parameters must always be provided as plain values or missed out to use the defaults.

Post-processing functions can be chained together in a comma-separated list, in which case they are applied in turn, each taking the output of the previous one. For example, the following query follows from all ssh processes to processes they are communicating with:

```
SEARCH DiscoveredProcess WHERE cmd HAS SUBWORD "ssh"
PROCESSWITH communicationForProcesses,
    localToRemote,
    processesForCommunication
```
After a search, you can immediately refine it.

```plaintext
SEARCH some search...
PROCESSWITH
where @2 = "foo"
show bar, baz, @3
```

@number refers to columns in the first search, as in current refine searches. Remember that column index starts with 0. It is very rare for this kind of refine search to be necessary. Normally, the same results are obtained more efficiently by simply performing the where clause directly in the search, avoiding the PROCESSWITH altogether.

**Named node sets and set operations**

Sets resulting from searches and traversals can be given names, and then combined with set operations. You name the node set using the AS keyword. The location in which you do this is important. It must appear after the node kind, or after the traversal if one is used in the SEARCH query. For example, to find Windows hosts running Oracle products:

```plaintext
SEARCH Host AS windows_hosts WHERE os HAS SUBWORD 'Windows'
SEARCH SoftwareInstance WHERE type HAS SUBWORD "oracle" TRAVERSE :::Host AS oracle_hosts
SEARCH IN (windows_hosts AND oracle_hosts) SHOW name, os
```

Valid operators for the sets are

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>and</td>
<td>Intersection of sets.</td>
</tr>
<tr>
<td>or</td>
<td>Union of sets.</td>
</tr>
<tr>
<td>-</td>
<td>Negative intersection.</td>
</tr>
<tr>
<td>(minus sign)</td>
<td>Negative intersection.</td>
</tr>
</tbody>
</table>

⚠️ **Many post-processing functions do not create node sets**

These set operations work only on sets of nodes. Many of the post-processing functions described below return results in which the rows do not correspond directly with nodes. Attempting to use such post-processed results in set manipulation operations leads to an error.

**Data manipulation functions**

The following functions manipulate data:
bucket(interval, min=0, max=0)

Separates data into ‘buckets’ representing ranges of values present. It looks at the first attribute selected with the SHOW clause, which should be either a number or a date. It starts by dividing the data into buckets with ‘width’ specified by `interval`. i.e. with an interval of 10, the first bucket contains the values between 0 and 10, the next between 10 and 20, and so on. The result then contains one row for each bucket, with two columns showing the bucket value and the number of input values in the bucket.

If provided, `min` and `max` specify the minimum and maximum number of buckets. If the number of buckets based on the `interval` is outside those boundaries, the interval is divided or multiplied by two until it fits. If used to group dates the interval is assumed to be in seconds. This function is mostly used to build charts.

```sql
SEARCH DiscoveryAccess
SHOW discovery_duration_sum
PROCESS WITH bucket(30,3,20)
```

unique(sort=0)

The `unique` function takes the rows of output from the search, and returns each unique row just once.

If the optional sort argument is set to 1, the result rows are sorted; if set to zero or not provided, the rows are output in the same order they appeared in the original results, with duplicates removed.

```sql
countUnique(show_node_count=1, show_total_count=1, sort=1, set_headings=None, ignore_none=1)
```

The `countUnique` function can be used, for example, to produce a summary of discovered processes:

```sql
SEARCH DiscoveredProcess SHOW cmd, args PROCESS WITH countUnique(0)
```

It converts the search results into a summary where each row contains a command, arguments pair and count of how many times that pair appears.

When a single list attribute is provided in the SHOW clause, `countUnique` counts the individual items in the list and shows two counts. The first count is the number of nodes in which the item appears; the second count is the total number of times the item appears, which will be a larger number if an items appears more than once in a list.

`countUnique` takes two optional boolean arguments to indicate which count columns appear.
If \textit{sort} is 1, \textit{countUnique} sorts the results by total; if it is set to zero, it keeps them in the order it found them in the original result set.

All boolean arguments default to 1, meaning both count columns appear and the results are sorted.

If \textit{set\_headings} is provided, it contains a list of strings to use as the headings in the result, overriding the default headings. The list must have the correct number of items corresponding for the number of columns (2 or 3 depending on the show parameters).

By default, values that are None are ignored; if \textit{ignore\_none} is set to zero, None values are counted in the same way as other values.

displayHistory(start, end, attr\_count)

The displayHistory function explodes a result set with history information.

```sql
SEARCH Host
SHOW name, name, ram, processor
PROCESS WITH displayHistory(parseTime("2009-01-01"), currentTime(), 1)
```

The three arguments are the start date, the end date and the number of attributes to leave as is. By default the end date is now and the attribute count is 1 so this invocation could be simplified as:

```sql
SEARCH Host
SHOW name, name, ram, processor
PROCESS WITH displayHistory(parseTime("2009-01-01"))
```

This example returns a result set with one line for every change in the history of the attributes name, ram and processor, detailing on each line the date it happened, and the attribute name and the attribute value before and after the change.

Provenance functions

These functions analyze provenance information:

provenanceDetails(friendly\_time=0)

provenanceDetails takes all of the attributes selected in the \textit{SHOW} clause and finds the provenance information for them. For each attribute it shows a row in the output containing the source node label, the attribute name, the attribute value, the time the attribute was last confirmed, and the label of the evidence node.

If the optional \textit{friendly\_time} argument is set to 1, the times are converted to strings; if set to 0 or not provided, times are returned in the internal time format.
provenanceFailures(friendly_time=0)

provenanceFailures is currently only supported for Host nodes. For each Host, it traverses to find the most recent DiscoveryAccess. It then finds the provenance information for each of the attributes in the SHOW clause. The output consists of one row for each attribute that was not confirmed in the most recent DiscoveryAccess. The rows contain the label of the Host node, the attribute name, the attribute value, the label of the evidence node used to set the attribute, the time the value was confirmed, and the time of the most recent DiscoveryAccess.

If the optional friendly_time argument is set to 1, the times are converted to strings; if set to 0 or not provided, times are returned in the internal time format.

Network connection functions

These functions analyze network connection data:

communicationForProcesses(targets=3, show="SUMMARY")

Given an input of DiscoveredProcesses, returns a list of node sets which varies depending on the value of targets:

<table>
<thead>
<tr>
<th>targets</th>
<th>return</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DiscoveredNetworkConnections</td>
</tr>
<tr>
<td>2</td>
<td>DiscoveredListeningPorts</td>
</tr>
<tr>
<td>3</td>
<td>both</td>
</tr>
</tbody>
</table>

Returns network connections and listening ports that tie up with the given set of processes. (For example, the network connection or listening port comes from the same discovery access as the process, and the process ids match.)

The show clause determines which attributes on the nodes are returned in tabular results. The same show clause is used for both node kinds.

The result of this function is useful for feeding in to the localToRemote or communicationToRemoteHost functions.

processesForCommunication(show="SUMMARY")

The input must contain DiscoveredNetworkConnections or DiscoveredListeningPorts or both. Returns a node set of DiscoveredProcesses.

Returns processes that tie up with the given network connections and listening ports. (For example, the processes come from the same discovery access as the connections and ports, and the process ids match.)

The show clause determines which attributes on the nodes are returned in tabular results.
localToRemote(targets=15, show="SUMMARY")

The input must contain DiscoveredNetworkConnections or DiscoveredListeningPorts, or both. Returns nodes corresponding to the 'other end' of the input communication information. The results depend upon the targets specification. targets is a 'bit mask' formed by adding together the numbers corresponding to the required results:

<table>
<thead>
<tr>
<th>targets</th>
<th>result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>remote DiscoveredNetworkConnections</td>
</tr>
<tr>
<td>2</td>
<td>remote DiscoveredListeningPorts</td>
</tr>
<tr>
<td>4</td>
<td>in-machine DiscoveredNetworkConnections</td>
</tr>
<tr>
<td>8</td>
<td>in-machine DiscoveredListeningPorts</td>
</tr>
</tbody>
</table>

The function distinguishes between 'remote' connections that are on different computers to the source information and 'in-machine' connections that are communication from one process to another on a single computer.

DiscoveredNetworkConnection nodes can be used to find both target DiscoveredNetworkConnection and DiscoveredListeningPort nodes.

DiscoveredListeningPort nodes can only be used to find target DiscoveredNetworkConnection nodes.

In all cases, only network connections and listening ports found during the most recent complete DiscoveryAccess are considered, meaning that only 'current' data is used.

It is not an error to, for example, pass in only DiscoveredListeningPorts and set targets to 2. In this instance, an empty list results.

The show clause determines which attributes on the nodes are returned in tabular results. The same show clause is used for both node kinds.

The result of this function is useful for feeding in to the processesForCommunication function.

hostToHostCommunication(show="SUMMARY")

Given a set of Host nodes, returns a set of Hosts that are communicating with those hosts, according to observed network connections.

A host is considered to be communicating with another if there is a network connection from either host to the other, according to the remote IP address on the network connection and the IP addresses of the host.

The show clause determines which attributes on the nodes are returned in tabular results.
communicationToRemoteHost(show="SUMMARY")
Given a list of node sets, which must contain DiscoveredNetworkConnections or DiscoveredListeningPorts (or both), return a set of Host nodes that are communicating.

A Host is returned if one of its IPs matches one of the remote IP addresses of one of the network connections, or if it has a network connection with a remote IP address and port that matches one of the listening ports.

The show clause determines which attributes on the nodes are returned in tabular results.

hostToRemoteCommunication(show="SUMMARY")
Given a set of Host nodes, returns a list of DiscoveredNetworkConnections and DiscoveredListeningPorts that are communicating with the hosts. Network connections are returned if their remote IP address corresponds to an IP address of one of the hosts, and listening ports are returned if a network connection on one of the hosts contains a remote IP address and the port matches the listening port.

Only network connections and listening ports found during the most recent complete DiscoveryAccess are considered, meaning that only ’current’ data is used.

The show clause determines which attributes on the nodes are returned in tabular results.

The result of this function is useful for feeding in to the processesForCommunication function.

communicatingSIs(show="SUMMARY")

Given a set of SoftwareInstance nodes, returns a set of the SoftwareInstance nodes with which they are communicating.
This function is equivalent to a traversal from SoftwareInstance to DiscoveredProcess, then a chain of communicationForProcesses, localToRemote, processesForCommunication, followed by a traversal from DiscoveredProcess back to SoftwareInstance.

In BMC Atrium Discovery version 8.2.01 and later, communicatingSIs returns SoftwareInstance nodes corresponding to both remote and in-machine communication links. Earlier versions only returned SoftwareInstance nodes on remote computers.

connectionsToUnseen
Takes a set of DiscoveredNetworkConnection nodes, and filters it to include only those ones that represent connections to addresses that have not been seen on IPAddress nodes present in the system. That is, it returns network connections to devices that have not been scanned.
networkConnectionInfo

Takes a set of DiscoveredNetworkConnection nodes and produces a summary of information about each one. Each row contains the name of the "local" Host, the associated local process command and arguments, port and IP address information for the connection, the remote command and arguments, and the remote host name. Not all of the information is always available (for example, if the remote side of the connection had not been scanned, or if insufficient discovery permission meant ports were not associated with processes).

networkConnectionInfo takes four optional list parameters to modify the attributes shown for Hosts and DiscoveredProcesses. The first two parameters specify the attributes to show on Host nodes and the headings to use for those columns; the second two parameters specify the attributes to show on DiscoveredProcess nodes and the headings to use for those. The number of headings must match the number of attributes shown for each node kind. In both cases, the attributes and headings are used twice, once for the local end of the network connection, and once for the remote end. The headings are prefixed "Local" and "Remote" as appropriate.

The default settings are equivalent to networkConnectionInfo(["name"], ["host name"], ["cmd", "args"], ["command", "arguments"]). It is valid to use key expressions in the attribute lists, so attributes from nodes related to the hosts or processes can be displayed.

siCommunicationSummary

Takes a set of SoftwareInstance nodes and produces a summary with one row for each observed network connection, treating the input Software Instance nodes as the local end. Each row contains the local Host and Software Instance, local and remote IP address and port, remote Software Instance and remote Host. Values for the remote end might not be available if the remote host was not scanned, or if the connection could not be associated with a particular Software Instance.

Like networkConnectionInfo, siCommunicationSummary takes four optional list parameters to modify the attributes shown for Hosts and Software Instances. The default is equivalent to siCommunicationSummary(["name"], ["host name"], ["name"], ["SoftwareInstance"]). Key expressions are valid in the attribute lists, so attributes from nodes related to the Hosts or Software Instances can be shown.

hostConnectionSummary

Takes a set of Host nodes and produces a summary with one row for each observed network connection, treating the input Host nodes as the local end. Each row contains the local Host, local and remote IP address and port, local and remote process using that remote connection (if available), and remote Host. The remote Host might not be available if it was not scanned, or if the connection could not be associated with a particular Host.
Chart-specific functions

These functions are only useful as input to charts:

timeInterval(column_count = 20, how_far_back = 0)

This function transforms a list of result sets with two dates into a result set of counts. An example usage is to graph the count of hosts over time. column_count determines how many data points will be present in the output, how_far_back is a time — the function will drop anything in the given set older that this date.

The function assumes that the first two columns in the passed set are dates and if another column is present returns one data set per possible value, somewhat similar to countUnique.

This function returns multiple result sets, one with a legend whose metadata dictionary contains an isLegend key. If there are more than two columns in the sources sets it returns one result set per unique value of this column:

- The result set title is set to the string version of the value.
- value in the metadata is the original CORBA any containing of the split value.
- index in the metadata preserves the order in which the split values were found in the source nodeset.

If there are only two columns, only two result sets are returned: the legend and a list of counts, and no metadata is set.

For example:

```
SEARCH FLAGS(include_destroyed) Host
   SHOW creationTime(#), destructionTime(#)
PROCESS WITH timeSeries()
```

A Note on Performance

There are often several ways to write a query that produces a required output report. Writing queries in different ways can sometimes cause enormous differences in query performance, so this section provides some hints on how to write queries so they are as efficient as possible.

Use of indexes

Wherever possible, use HAS SUBWORD rather than MATCHES/LIKE or HAS SUBSTRING so the datastore indexes are used. The full text index is very good at finding words and phrases.

WHERE clause ordering

In WHERE clauses containing AND expressions, order the parts so that the most restrictive condition comes first, except that it is always better to use a HAS SUBWORD condition in preference to other kinds. This reduces the search space as soon as possible.
Filtering on related nodes

To filter nodes by characteristics of related nodes, it is better to find the related nodes, then traverse to the required nodes. For example, to find Hosts in London, this query finds the London location in the index then uses a traversal so it directly and quickly finds the required Hosts:

```
SEARCH Location
WHERE name = "London"
TRAVVERSE Location:Location:ElementInLocation:Host
```

This query starts by building a set of all Host nodes, then checks the location of each one, which is much slower:

```
SEARCH Host
WHERE NODECOUNT(TRAVERSE Location:Location:ElementInLocation:Host
  WHERE name = "London")
```

This final version also builds a set of all Hosts and checks the location of each one, but in this case it fails when encountering Hosts in more than one location, as well as being inefficient. Queries like this should never be used:

```
SEARCH Host
WHERE #Location:Location:ElementInLocation:Host.name = "London"
```

Default ordering

By default, queries with no `SHOW` clause are ordered by the label value defined in the taxonomy. If the data is to be exported to an external system that does not care about order, the time spent retrieving data to sort can be avoided by specifying `ORDER BY ""`.

Context-sensitive reporting and linking

This section explains how to create context-sensitive reports and links with BMC Atrium Discovery.

- Configuring additional links
- Custom reporting
- Managing channels
Configuring additional links

The external links feature provides links from objects modelled in BMC Atrium Discovery, to other web-based systems where additional information is stored. It is also used to create links to reports or charts specified for individual nodes or sets of nodes (result sets).

For example, if you are viewing a host in the BMC Atrium Discovery UI, you might see a link to that host in a change management application such as Remedy. See Dynamic Toolbox. Search queries are used to construct the links. See Using the Search and Reporting service for more information.

The following link types are provided:

- Links to external systems
- Report for individual nodes
- Chart for individual nodes
- Report for result sets
- Chart for result sets

Configuration files

Additional link configuration has changed in version 7.3 to become a form of Custom reporting. The xml configuration files are held in the following directories:

1. /usr/tideway/data/installed/reports/
2. /usr/tideway/data/custom/reports/

The directories are parsed in the order given (installed before custom), and the files contained in these directories are parsed in alphabetical order, with numbers before letters. This order is important as later definitions for a named link override those loaded earlier. The standard additional links are contained in /usr/tideway/data/installed/reports/00additional_links.xml.

Upgrades from previous versions remove /usr/tideway/data/default/additional_links.xml and add the following new file: /usr/tideway/data/installed/reports/00additional_links.xml

If you are upgrading, you must copy /usr/tideway/data/customer/additional_links.xml to /usr/tideway/data/custom/reports/00additional_links.xml and manually convert the file to the new format which is described on this page.

Links configured in the installed subdirectory are displayed beneath the default links, those from the custom subdirectory follow them. However, if a custom link overrides an installed link it will appear in place of the installed link.
Notes

1. The link must be specified in valid XML. The XML equivalents of certain characters must be used (for example, `&amp;` instead of `, `&gt;` instead of `, and `&lt;` instead of `<`).
2. The XML files are read when the Application server starts. Usually you would restart the whole tideway service.
3. Parse errors in the XML do not prevent the Application server starting but might cause one or all external link definitions to be ignored.
4. Failure to return a value for a requested attribute will be caught, resulting in the link being discarded for this particular node, on this request. For future requests an attempt to build the link will be made again.
5. You can substitute any attribute in a tag. Use `%(attribute_name)s` to do this. If the attribute contains a list value there will also be a string representation containing a comma separated list of values in `%(comma_separated_attribute_name)s`.
6. The following keys are provided:
   - `id`: the node ID as set in the URL, for example `nodeID=%(id)s`
   - `kind`: the node kind.
   - `uiModule`: the UI module that normally handles display of the node kind.
   - `urlPrefix`: the URL prefix of the appserver. This is usually `ui`.
7. In addition `<link>` with `<attribute>` entries contain:
   - `translated`: the locale translated attribute name.
   - `otherNodeID`: if the attribute represents a link then the node ID of the related node.

Link tags

The link tag specifies the type of link that is being created. These can be one of the following:

- **Normal Link** — specified with a `<link>` tag.
  A report that is run against a single node and the link should be visible when viewing a node of the correct node kind.
- **Chart Link** — specified with a `<node-chart>` tag.
  A chart that is run against a single node. The link should be visible when viewing a node of the correct node kind.
- **Set Report Link** — specified with a `<refine-report>` tag.
  A report that is run against a nodeset and the link should be visible when viewing a set of nodes of the correct node kind.
- **Set Chart Link** — specified with a `<refine-chart>` tag.
  A chart that is run against a nodeset and the link should be visible when viewing a set of nodes of the correct node kind.

Each tag must specify a name using the `name` attribute. Each link tag is in a distinct namespace.
Tags available

The following xml tags are available in all form of links:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
</table>
| Title | Provides the main page title for the link. This tag is mandatory.  
<title>View trouble ticket for %hostname%</title> |
| Description | Populates the title attribute of the HTML link. Presented as a tool tip. Also shown as a subtitle below the main page title. This tag is mandatory.  
<description>Shows the distribution of OS type on all hosts</description> |
| Kind | Link will be shown against this node kind. This tag is mandatory.  
<kind>Host</kind> |
| Image | Specifies an image to display. The image should already exist on the appliance as the specification is from the appliance image root.  
{{http://appliance/styles/default/images/image_name}}. This tag is mandatory.  
<image>icon.gif</image> |
| Permissions | Specifies a list of permissions that the user must possess in order to display the link. <permissions> contains a list of <permission> entries, one per required permission.  
<permissions><permission>PERM_REPORTS_WRITE</permission><permissions> |

The following xml tags are available in <link>:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
</table>
| URL  | The URL that the link points to. This tag is mandatory.  
{{http://tickets.tideway.com}} |
| Attribute | Displays a link in the UI next to the attribute if the attribute is not displayed, or not present on the node, then the link is not displayed.  
<attribute>hostname</attribute>  
Relationships can be specified by concatenating role, relationship, role and destination node kind with _.  
<attribute>Previous_EndpointIdentity_Next_Host</attribute> |
| Target | Populates the HTML anchor for the link. Points to the named window or frame that the results of this query will be displayed in. Frequently this will be _new so that the link will open in a new window.  
<target>_new</target> |

The following xml tags are available in the remaining link types:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
</table>
| Lookup | Supported in <node-chart> and <node-report>. Indicates if the query is relative to the node. Defaults to true which will generate a search of the form LOOKUP '%id%'. If you need a general search then use this tag to override the behaviour.  
<lookup>False</lookup> |
| Flags | Any flags to include in the search query.  
<flags>include_destroyed</flags> |
| With |
### Tag Details

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any with functions to evaluate in the search query.</td>
<td><code>&lt;with&gt;</code>value(NODECOUNT(TRAVERSE Host:HostedSoftware:RunningSoftware:SoftwareInstance)) AS si_count&lt;/with&gt;`</td>
</tr>
<tr>
<td>Where</td>
<td>Any conditions in the search query. This tag takes an optional keyword attribute which, when given value False, prevents the addition of the WHERE keyword when constructing the query. For example, TRAVERSE :::Host must not be prefixed by WHERE. <code>&lt;where&gt;version = ''&lt;/where&gt;</code></td>
</tr>
<tr>
<td>Order-by</td>
<td>Sort results of query by specified order. <code>&lt;order-by&gt;</code>name&lt;/order-by&gt;`</td>
</tr>
<tr>
<td>Show</td>
<td>List of attributes to display. For <code>&lt;node-chart&gt;</code> and <code>&lt;refine-chart&gt;</code> this represents the list of attributes shown when clicking through the chart. <code>&lt;show&gt;</code>SUMMARY&lt;/show&gt;`</td>
</tr>
<tr>
<td>Split</td>
<td>Supported in <code>&lt;node-chart&gt;</code> and <code>&lt;refine-chart&gt;</code>. Indicates the column to chart. This tag takes an optional columns attribute which specifies the number of columns generated by the split. For example <code>&lt;split columns=&quot;2&quot;&gt;endtime PROCESSWITH bucket(3600, 3, 48)&lt;/split&gt;</code> retrieves the endtime and puts the values into a list of buckets producing 2 columns, the time and the count. The title of the column is used as the axis title. A maximum of 2 <code>&lt;split&gt;</code> tags can be defined, the first for the x-axis, the second the y-axis. <code>&lt;split&gt;os AS 'Detailed OS'&lt;/split&gt;</code></td>
</tr>
<tr>
<td>Y-axis-title</td>
<td>Supported in <code>&lt;node-chart&gt;</code> and <code>&lt;refine-chart&gt;</code>. Indicates the title of the y axis column. This will default to 'Count' unless the <code>&lt;split&gt;</code> provides a second column. <code>&lt;y-axis-title&gt;Hosts&lt;/y-axis-title&gt;</code></td>
</tr>
</tbody>
</table>

The default type of chart can be specified by adding the default attribute to `<node-chart>` or `<refine-chart>`. Supported charts depend upon the number of columns specified by `<split>`.

For single columns the supported charts are:

- pie (default)
- bar
- column

For two columns the supported charts are:

- line (default)
- column

**Example 00additional_links.xml file**

An example XML file is broken down and its constituent parts are described below:

All files start with an XML declaration and an `<reports>` tag.

```xml
<?xml version="1.0"?>
<reports version="2.0">```
The reports version must be 2.0.

**External links**

Each external link is defined under a `<link>` tag.

The following example shows a label populated with the text *View trouble ticket for hostname*, Where *hostname* is the name of the current host.

The URL in the example is to a trouble-ticketing system with a web interface. This assumes that there is a host attribute called ticketNum which contains the trouble ticket number.

```xml
<link name="Link.Host.Ticket">
  <title>View trouble ticket for %(hostname)s</title>
  <description>Browse trouble ticket system for this host.</description>
  <kind>Host</kind>
  <image>kind/gif/Location_16.gif</image>
  <url>http://tickets.tideway.com/tickets/%(ticketNum)s.cgi</url>
</link>
```

**Report links**

Each report link is defined under a `<node-report>` tag.

The following examples retrieves a list of hosts with the same OS type as the currently viewed host.

```xml
<node-report name="Node.Report.Host.SameOS">
  <title>Hosts with the same OS type</title>
  <description>List of hosts with the same OS type</description>
  <kind>Host</kind>
  <image>system/gif/report_16.gif</image>
  <lookup>False</lookup>
  <where>os_type = %(os_type)r</where>
</node-report>
```

**Chart links**

Each chart link is defined under a `<node-chart>` tag.

A chart showing access methods for this host node.

```xml
<node-chart name="Node.Chart.Host.AccessMethod" default="column">
  <title>Host Login Summary Chart</title>
  <description>Host Login Methods on a column chart</description>
  <kind>Host</kind>
  <image>system/gif/column_chart_16.gif</image>
</node-chart>
```
SetReport links
Each set report link is defined under a `<refine-report>` tag.

Show the number of hosts per software package on which the currently viewed set of packages are installed.

```xml
  <title>Package Host Density</title>
  <description>How many Hosts are these Packages installed on</description>
  <kind>Host</kind>
  <image>system/gif/report_16.gif</image>
  <show>
    name,
    version,
    NODECOUNT(TRAVERSE InstalledSoftware:HostedSoftware:Host:Host)
    AS '_|Number of Hosts|_'
  </show>
</refine-report>
```

SetChart links
Each set chart link is defined under a `<refine-chart>` tag.

Summarise the currently viewed set of Software Instance nodes in a pie chart segmented by type.

```xml
<refine-chart name="Refine.Chart.Host.SISummary" default="pie">
  <title>Software Instance Type Summary</title>
  <description>Show a Summary of the Software Instance Types</description>
  <kind>Host</kind>
  <image>system/gif/pie_chart_16.gif</image>
  <order-by>type</order-by>
  <split>type</split>
  <show>SUMMARY</show>
</refine-chart>
```

Completion
Finally, the XML file is completed with the closing tag.

```xml
</reports>
```
Custom reporting

The custom reporting feature enables you to perform customization of the reports and channels on the appliance. Channels may be created more simply using the Channels page.

Configuration files

The reporting facilities and selection of UI channels shown on each page are configured using xml configuration files which are held in the following directories:

1. /usr/tideway/data/installed/reports/
2. /usr/tideway/data/custom/reports/

The directories are parsed in the order given (installed before custom), and the files contained in these directories are parsed in alphabetical order, with numbers before letters. This order is important as later definitions for a named report override those loaded earlier. The standard reports are contained in /usr/tideway/data/installed/reports/00reports.xml.

The configuration is read when the UI process starts. Changes to the report configuration files require a UI restart. Errors in the configuration files do not prevent the Application server from starting but might cause various reports and channels to be missing.

To get the UI to update with your custom reports, do the following:

1. log out of the UI
2. from the command-line, run this command: sudo /sbin/service tideway restart appserver
3. log in to the UI

⚠️ You cannot add context sensitive reports to the discovery status page.

You cannot customize the Host Profile report.

The root element of each reports.xml file must be <reports version="2.0">. The <reports> element can have several types of elements:

- Standard reports:
  - <chart>: a chart. See <chart> elements.
- Channels:
  - <chart-channel>: a simple chart channel. See <chart-channel> elements.
  - <chart-multi-channel>: a chart channel with multiple values per column. See <chart-multi-channel> elements.
• `<report-channel>`: a channel containing a list of charts and reports. See `<report-channel>` elements.
• `<rss-channel>`: a rss feed channel. See `<rss-channel>` elements.
• `<summary-channel>`: a channel containing node kind counts. See `<summary-channel>` elements.
• `<time-series-channel>`: a chart channel containing values over time. See `<time-series-channel>` elements.
• `<video-channel>`: a video channel. See `<video-channel>` elements.
• `<web-channel>`: a channel containing a web feed. See `<web-channel>` elements.

• Page information:
  • `<page>`: the selection of channels on one page. See `<page>` elements.

**<report> elements**

The `<report>` element was extended in version 7.3 by merging in functionality from `<extrareport>` which no longer exists. A `<report>` element requires a `name` attribute giving a unique name to the report.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Provides the main page title for the report. This tag is mandatory.</td>
</tr>
<tr>
<td>Description</td>
<td>Populates the title attribute of the HTML link. Presented as a tool tip. Also shown as a subtitle below the main page title. This tag is mandatory.</td>
</tr>
<tr>
<td>Kind</td>
<td>Kind to use in search. Multiple kinds can be specified in a comma separated list or by using the * wildcard.</td>
</tr>
<tr>
<td>Flags</td>
<td>Any flags to include in the search query. This tag takes an optional keyword attribute which, when given value False, prevents the addition of the FLAGS( prefix and ) postfix. The attribute is normally used if parameters are used control the flags.</td>
</tr>
<tr>
<td>With</td>
<td>Any with functions to evaluate in the search query.</td>
</tr>
<tr>
<td>Where</td>
<td>Any conditions in the search query. This tag takes a couple of options attributes:</td>
</tr>
<tr>
<td></td>
<td>• keyword which, when given value False, prevents the addition of the WHERE keyword when constructing the query. For example, TRAVERSE ::Host must not be prefixed by WHERE.</td>
</tr>
<tr>
<td></td>
<td>• start which, when given value False, places the parameter conditions at the start of the query rather than the end. Normally parameter queries are added after the <code>where</code> text. However, if there is a need to perform tests before a traversal then they can be placed at the beginning.</td>
</tr>
<tr>
<td>Order-by</td>
<td>Sort results of query by specified order.</td>
</tr>
</tbody>
</table>
**Tag Details**

**Show**
List of attributes to display. This tag takes an optional `keyword` attribute which, when given value `False`, prevents the addition of the `SHOW` keyword.

```xml
<show name, os, #InferredElement:Inference:Primary:HostInfo.hostid</show>
```

**Url**
Redirect browser when report is selected.

```xml
<url>NetworkMismatchSummaryReport</url>
```

**Imports**
List of `<import>` each of which contains a Python module to import. These are used by the `<parameters>` for `<convert>`, `<default>` and `<eval>` described below.

```xml
<imports>
  <import>common.timeutil</import>
</imports>
```

**Parameters**
List of `<parameter>` each of which contains an interactive parameter filled in by the user. These are described in detail below.

---

### `<parameter>` element

Interactive parameters are defined using `<parameter>` elements. They must have a `name`, representing the name of the parameter, that is unique in the report.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Label of the parameter shown when it is rendered in a form. This tag is mandatory.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of control widget to use for the parameter in the form and is detailed below. This tag is mandatory.</td>
</tr>
<tr>
<td>Key</td>
<td>References a replacement construct in the constructed query.</td>
</tr>
<tr>
<td>Where</td>
<td>Query fragment to add to where clause of the parent element's search string. The reports system inserts the value of the parameter using string substitution with a key of <code>value</code>.</td>
</tr>
</tbody>
</table>

When the query is built from parameters all the parameters are evaluated as specified by `<type>`. If a parameter has a where clause it is substituted into the where clause with a key of `value`. All parameters without a key are then joined together using `AND` and inserted either at the beginning (`<where start="True">` specified) or end of the where clause in the report. The complete query is then constructed. If there are any parameters with a key the completed query has those parameters inserted using string substitution with the specified key of the parameter.

The `<type>` tag must have a `name` attribute which specifies the type of the parameter. It should have one of the following values:

- **TextField**: a text entry field.
- **SelectField**: a list.
- **RelationshipSearchField**: a popover node-selection field.
- **DateTimeField**: a time field.

The tags that can be used with `<type>` depend upon these values.
<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>If type is SelectField add an <em>All</em> option to the select list (the default). When given value False prevents the addition of the 'All' option. &lt;all&gt;False&lt;/all&gt;</td>
</tr>
<tr>
<td>Convert</td>
<td>Function used to convert the default value and user-typed value for use in query. For example, the function could be <code>common.timeutil.convertFromUnix</code>. This will convert the value to UNIX form, which can then be added to the query. Referenced functions should be fully specified, and the packages and modules required should appear in <code>&lt;imports&gt;</code> elements, described above. Functions are referenced rather than called so do not need parentheses. The function can also be a lambda function taking one parameter. &lt;convert&gt;common.timeutil.convertFromUnix&lt;/convert&gt;</td>
</tr>
<tr>
<td>Default</td>
<td>Expression that defines a default value for the parameter. Referenced functions should be fully specified, and the packages and modules required should appear in <code>&lt;imports&gt;</code> elements, described above. &lt;default&gt;common.timeutil.convertToUnix(common.timeutil.currentTime())&lt;/default&gt;</td>
</tr>
<tr>
<td>Escape</td>
<td>Escape string value. By default string values are escaped. When given value False prevents the escaping. &lt;escape&gt;False&lt;/escape&gt;</td>
</tr>
<tr>
<td>Kind</td>
<td>If type is RelationshipSearchField is used to determine which source kind when finding a list of relationships defined in taxonomy. &lt;kind&gt;Host&lt;/kind&gt;</td>
</tr>
<tr>
<td>Size</td>
<td>If type is TextField specifies the size of the text field otherwise specifies the number of visible options in the list. &lt;size&gt;2&lt;/size&gt;</td>
</tr>
</tbody>
</table>
| Options | If type is SelectField then there are several ways to specify the selection options:  
  
  • `eval` which evaluates the expression. Referenced functions should be fully specified, and the packages and modules required should appear in `<imports>` elements, described above. <eval>api.audit.getEvents()</eval>  
  
  • `option` which contains the option text. The `<option>` takes an optional `value` attribute which is used when the option is selected. If no `value` attribute is present the option text is used. Multiple `<option>` values can be provided. <option value="21">FTP</option>  
  
  • `query` which performs a search query the first column of which is used as selection values. `<query>SEARCH Location SHOW name</query>`  
  
  Only one of the selection option forms can be specified. |
| Validate | How to validate the parameter. Currently supports:  
  
  • `integer`: value is an integer  
  • `number`: value is a number  
  • `boolean`: value is a boolean  
  • `NotEmpty`: value is not empty  
  
  If the value does not validate the form will present an error to the user. <validate>NotEmpty</validate> |

Bringing all this together, the following element defines a parameter which appears in the UI labelled *Type* as a list with three specified values and no *All* entry. The `<where>` tag means that should the user choose *Host* then `kind(#) = 'Host'` will be added to the report query.
The following report will count the number of instances of software instances on host optionally limiting the software instances to a particular type chosen by the user from a list.

<report name="Software.Report.InstanceSummary">
  <title>Software Inventory</title>
  <description>Shows summary of distribution of a piece of software</description>
  <kind>SoftwareInstance</kind>
  <show>
    type AS "|Type|",
    product_version AS "|Product Version|"
    PROCESS WITH
    countUnique(1,0)
  </show>
  <parameters>
    <parameter name="Software_Report_InstanceSummary_type">
      <title>Product type</title>
      <type name="SelectField">
        <query>
          SEARCH SoftwareInstance
          ORDER BY type
          SHOW type
          PROCESS WITH unique()
        </query>
      </type>
      <where>type = %(value)s</where>
    </parameter>
  </parameters>
</report>

<chart> elements
The <chart> element in fundamentally the same as the <report> element with the following changes:
the default type of chart can be specified by adding the `default` attribute to `<chart>`,
the `<show>` tag is used to determine the click through columns of the chart.

### Additional tags are:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
</table>
| Split     | Indicates the column to chart. This tag takes an optional columns attribute which specifies the number of columns generated by the split. For example,  
```xml<br>  <split columns="2">endtime PROCESSWITH bucket(3600, 3, 48)</split><br>```retrieves the endtime and puts the values into a list of buckets producing 2 columns, the time and the count. The title of the column is used as the axis title. A maximum of 2 `<split>` tags can be defined, the first for the x-axis, the second the y-axis.  
```xml<br>  <split>os AS 'Detailed OS'</split><br>``` |  
| Y-axis-title | Indicates the title of the y axis column. This will default to Count unless the `<split>` provides a second column.  
```xml<br>  <y-axis-title>Hosts</y-axis-title><br>``` |  

For single columns the supported charts are:

- pie (default)
- bar
- column

For two columns the supported charts are:

- line (default)
- column

The following chart will show a count of different OS classifications. By default a pie chart will be shown and when clicked through a summary report for the appropriate hosts. When displaying bar and column charts the y-axis will be labelled Hosts.

```xml<br>  <chart name="Infrastructure.Chart.OSClassification" default="pie"><br>  <title>Host OS Classification</title><br>  <description>Show count of Hosts for each OS Classification</description><br>  <kind>Host</kind><br>  <order-by>os_class</order-by><br>  <split>os_class AS "_[OS Class]_"></split><br>  <y-axis-title>"_Hosts_"</y-axis-title><br>  <show>SUMMARY</show><br>  </chart><br>```  

**<chart-channel> elements**  
The `<chart-channel>` element requires a `name` attribute giving a unique name to the chart channel. An optional `default` attribute gives the initial chart type.
The following channel shows a pie chart of software product categories:

```xml
<chart-channel name="Channel.SWCategory" default="pie">
  <title>Software Products By Category</title>
  <description>Shows a list of the Software by Category</description>
  <split>explode(#Element:Maintainer:Pattern:Pattern.categories) AS 'Software Category'</split>
  <kind>SoftwareInstance</kind>
  <show>
    summary,
    #Element:Maintainer:Pattern:Pattern.categories AS 'Software Category'
  </show>
</chart-channel>
```

**<chart-multi-channel> elements**

The `<chart-multi-channel>` element requires a `name` attribute giving a unique name to the chart multi channel. An optional `default` attribute gives the initial chart type.
<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kind</td>
<td>Kind to use in search. Multiple kinds can be specified in a comma separated list or by using the * wildcard.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;kind&gt;Host&lt;/kind&gt;</code></td>
</tr>
<tr>
<td>With</td>
<td>Any with functions to evaluate in the search query.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;with&gt;value(NODECOUNT(TRAVERSE Host:HostedSoftware:RunningSoftware:SoftwareInstance)) AS si_count&lt;/with&gt;</code></td>
</tr>
<tr>
<td>Where</td>
<td>Any conditions in the search query.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;where&gt;version = ''&lt;/where&gt;</code></td>
</tr>
<tr>
<td>Order-by</td>
<td>Sort results of query by specified order.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;order-by&gt;name&lt;/order-by&gt;</code></td>
</tr>
<tr>
<td>Split</td>
<td>Indicates the column to chart. The title of the column is used as the axis title. There must be 2 <code>&lt;split&gt;</code> tags.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;split&gt;os AS 'Detailed OS'&lt;/split&gt;</code></td>
</tr>
<tr>
<td>Show</td>
<td>List of attributes to display when click through chart.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;show&gt;name, os, #InferredElement:Inference:Primary:HostInfo.hostid&lt;/show&gt;</code></td>
</tr>
</tbody>
</table>

The following channel shows a column chart of database versions with a column for each product type. Each column contains a list of found versions.

```xml
<chart-multi-channel name="Channel.SWDBVersion" default="column_nolegend">
  <title>Database versions</title>
  <description>Number and version of all the Database instances</description>
  <split>type</split>
  <split>(product_version or 'Unknown')</split>
  <kind>Pattern</kind>
  <where>
    'Relational Database Management Systems' in categories
    TRAVERSE Pattern:Maintainer:Element:SoftwareInstance
  </where>
  <show>SUMMARY</show>
</chart-multi-channel>
```

**<report-channel> elements**

The `<report-channel>` element requires a name attribute giving a unique name to the report channel.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Channel title. This tag is mandatory.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;title&gt;Operating Systems&lt;/title&gt;</code></td>
</tr>
<tr>
<td>Description</td>
<td>Shown when editing the channels. This tag is mandatory.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;description&gt;Operating System Reports and Charts&lt;/description&gt;</code></td>
</tr>
<tr>
<td>Image</td>
<td>Image to show for channel.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;image&gt;discovery&lt;/image&gt;</code></td>
</tr>
</tbody>
</table>
Tag | Details
---|---
Chart name. The chart definition must appear before this reference. Multiple `<chart>` tags can be specified. `<chart>`Infrastructure.Chart.OSClassificationPie</chart>

The following channel contains a link to a report and chart.

```xml
<report-channel name="Infrastructure.Channel.NetworkPolicy">
  <title>Network Policies</title>
  <description>Shows a list of Network Infrastructure Reports</description>
  <chart>Infrastructure.Chart.IPAddressDistribution</chart>
</report-channel>
```

**<rss-channel> elements**

The `<rss-channel>` element requires a `name` attribute giving a unique name to the rss channel.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Channel title. This tag is mandatory. <code>&lt;title&gt;Operating Systems&lt;/title&gt;</code></td>
</tr>
<tr>
<td>Description</td>
<td>Shown when editing the channels. This tag is mandatory. <code>&lt;description&gt;Operating System Reports and Charts&lt;/description&gt;</code></td>
</tr>
<tr>
<td>Render-description</td>
<td>Flag if to show RSS feed description, defaults to False. <code>&lt;render-description&gt;True&lt;/render-description&gt;</code></td>
</tr>
<tr>
<td>Url</td>
<td>URL for RSS feed. This tag is mandatory. <code>&lt;url&gt;http://www.tideway.com/confluence/createrssfeed.action?types=page&lt;/url&gt;</code></td>
</tr>
</tbody>
</table>

The following channel shows an RSS feed for VMware.

```xml
<rss-channel name="Channel.RSS.VMware">
  <title>VMware Feed</title>
  <description>Shows the latest info from VMware</description>
  <url>http://vmware.simplefeed.net/rss/?f=2fe7e950-01de-11de-3b40-003048605010</url>
</rss-channel>
```

**<summary-channel> elements**

The `<summary-channel>` element requires a `name` attribute giving a unique name to the summary channel.
Tag | Details
--- | ---
Title | Channel title. This tag is mandatory.  
\[
\text{\textless title\textgreater Operating Systems\textless/-title}\]
Description | Shown when editing the channels. This tag is mandatory.  
\[
\text{\textless description\textgreater Operating System Reports and Charts\textless/description}\]
Image | Image to show for channel.  
\[
\text{\textless image\textgreater discovery\textless/image}\]
Kind-count | Node kind for which to show count. Multiple \textless kind-count\textgreater entries can be specified.  
\[
\text{\textless kind-count\textgreater Host\textless/kind-count}\]

The following channel shows the number of Business Application Instance and Software Instance nodes.

\[
\text{\textless summary-channel name=\textquoteright Applications.Channel.Summary\textquoteright }\>
\text{\textless title\textgreater Application Summary\textless/title\textgreater}
\text{\textless description\textgreater Shows a summary of Applications\textless/description\textgreater}
\text{\textless image\textgreater applications\textless/image\textgreater}
\text{\textless kind-count\textgreater BusinessApplicationInstance\textless/kind-count\textgreater}
\text{\textless kind-count\textgreater SoftwareInstance\textless/kind-count\textgreater}
\text{\textless/summary-channel\textgreater}
\]

\textbf{\textless time-series-channel\textgreater elements}

The \textbf{\textless time-series-channel\textgreater} element requires a \textit{name} attribute giving a unique name to the time series channel.

This is identical to \textbf{\textless chart-channel\textgreater} with two exceptions:

- It performs a historical search and partition the result into a number of buckets. By default, this is 5 but can be overridden by specifying the \textit{time-series} attribute on the \textbf{\textless time-series-channel\textgreater} element.
- \textbf{\textless split\textgreater} is optional and if not present the chart will be based on the number of the specified node kind rather than the value of the specified attribute.

The following channel shows the number of Hosts for each UNIX OS over time.

\[
\text{\textless time-series-channel name=\textquoteright Channel.OSUNIX\textquoteright time-series=\textquoteright 7\textquoteright default=\textquoteright line\textquoteright }\>
\text{\textless title\textgreater UNIX Operating Systems\textless/title\textgreater}
\text{\textless description\textgreater Shows a count of Hosts for each UNIX OS\textless/description\textgreater}
\text{\textless split\textgreater os_type as \textquoteright OS Version\textquoteright\textless/split\textgreater}
\text{\textless kind\textgreater Host\textless/kind\textgreater}
\text{\textless where\textgreater os_class = \textquoteright UNIX\textquoteright\textless/where\textgreater}
\]
<video-channel> elements

The <video-channel> element requires a name attribute giving a unique name to the video channel.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Channel title. This tag is mandatory. &lt;title&gt;Operating Systems&lt;/title&gt;</td>
</tr>
<tr>
<td>Description</td>
<td>Shown when editing the channels. This tag is mandatory. &lt;description&gt;Operating System Reports and Charts&lt;/description&gt;</td>
</tr>
<tr>
<td>Image</td>
<td>Image to show for channel. &lt;image&gt;discovery&lt;/image&gt;</td>
</tr>
<tr>
<td>Src</td>
<td>Video source. This tag is mandatory. &lt;src&gt;/videos/Prerelease3.swf&lt;/src&gt;</td>
</tr>
</tbody>
</table>

The following channel shows an video feed for creating host profiles.

```xml
<video-channel name="Channel.Video.HostProfiles">
  <title>Feature Tutorial: Host Profiles</title>
  <description>Video Tutorial that explains the Host Profiles feature</description>
  <src>/videos/HostProfiles.swf</src>
</video-channel>
```

<web-channel> elements

The <web-channel> element requires a name attribute giving a unique name to the web channel.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Channel title. This tag is mandatory. &lt;title&gt;Google&lt;/title&gt;</td>
</tr>
<tr>
<td>Description</td>
<td>Shown when editing the channels. This tag is mandatory. &lt;description&gt;Shows the Google search engine&lt;/description&gt;</td>
</tr>
<tr>
<td>URL</td>
<td>URL for the web feed. This tag is mandatory. &lt;url&gt;<a href="http://www.google.com">http://www.google.com</a>&lt;/url&gt;</td>
</tr>
</tbody>
</table>

The following channel shows a web feed for the BMC Atrium Discovery Community forum.

```xml
<web-channel name="Channel.Web.BMCCommunity">
  <title>BMC Atrium Discovery Community</title>
  <description>Web feed for the BMC Atrium Discovery Community forum</description>
  <url>http://community.bmc.com</url>
</web-channel>
```
The `<page>` element requires a `name` attribute giving a unique name to the page.

### Built-in channels

The following built-in channels are defined:

<table>
<thead>
<tr>
<th>Built-In Channel Name</th>
<th>Description</th>
</tr>
</thead>
</table>

### Built-in page names

The following are the identifiers for pages with configurable content:

<table>
<thead>
<tr>
<th>Built-in Page Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationHome</td>
<td>The front page of the applications module.</td>
</tr>
<tr>
<td>DiscoveryReports</td>
<td>Reports in the discovery module.</td>
</tr>
<tr>
<td>Home</td>
<td>The appliance home page.</td>
</tr>
<tr>
<td>InfrastructureHome</td>
<td>The front page of the infrastructure module.</td>
</tr>
<tr>
<td>Reports</td>
<td>The front page of the reports module.</td>
</tr>
</tbody>
</table>

### Managing channels

Channels are small, self-contained pieces of content that are displayed on various pages throughout BMC Atrium Discovery. For example, the “Infrastructure” page is composed of several channels, including the Infrastructure Summary channel and a channel displaying Virtualization and Partitioning General Reports. The Channels page enables you to edit existing channels and create new ones.
Dashboards, which are displayed on the home page, are made up of user-configurable sets of channels. You can see which dashboards are available by clicking the Available Dashboards icon in the dynamic toolbox on the Home page. To view a dashboard, select it from the displayed list.

There are a variety of types of content available for use in channels. For example, you can configure video content, web feeds, and RSS feeds, in addition to the standard BMC Atrium Discovery reports. For more information on dashboards and channels, see Using and customizing dashboards and Custom reporting.

To view the channels page

To view the Channels page, from the Administration tab, click Channels. The page shows a list of channels.

<table>
<thead>
<tr>
<th>Name</th>
<th>Displays a channel name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Displays a description for the channel.</td>
</tr>
<tr>
<td>Content</td>
<td>Displays the summary of what the channel contains.</td>
</tr>
<tr>
<td>Actions</td>
<td>Displays actions to be performed on the channel:</td>
</tr>
<tr>
<td>Preview</td>
<td>Click to view the content of the channel.</td>
</tr>
<tr>
<td>Edit</td>
<td>Click to edit the channel definition. This is only shown for channels which can be edited. For example, you cannot edit a built in chart.</td>
</tr>
<tr>
<td>Delete</td>
<td>Click to delete the channel definition. This is only shown for channels which you have created.</td>
</tr>
</tbody>
</table>

Adding a new channel

You can add various types of channels using this page:

- **Report Channel** — A channel containing a list of charts and reports. See To add a report channel.
- **Summary Channel** — A channel containing node kind counts. See To add a summary channel.
- **Video Channel** — A video channel. See To add a video channel.
- **Rss Channel** — A rss feed channel. See To add an RSS channel.

To add a report channel

You can add a report channel that contains standard BMC Atrium Discovery reports. To do this:
1. From the Channels page, click **Add Report Channel**.
   The Add Report Channel page is displayed:

   ![Add Report Channel](image)

   The Add Report Channel page displays the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the name for the channel. The name can only contain alphanumeric, underscore, and period characters.</td>
</tr>
<tr>
<td>Title</td>
<td>Enter the title to be displayed in the channel header</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description of the channel. It is displayed when editing the channel.</td>
</tr>
<tr>
<td>Reports</td>
<td>Select the reports to add to the channel from the list of available reports. Move the selected reports to or from the Selected Reports list using the left or right arrows. Reorder the reports in the Selected Reports list using the up or down arrows.</td>
</tr>
</tbody>
</table>

2. Select **Apply** to create the channel or **Cancel** to discard the changes and return to the channels page.

To add a summary channel

You can add a summary channel that shows node counts for selected node kinds. To do this:

1. From the Channels page, click **Add Summary Channel**.
   The Add Summary Channel page displays the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the name for the channel. The name can only contain alphanumeric, underscore, and period characters.</td>
</tr>
<tr>
<td>Title</td>
<td>Enter the title to be displayed in the channel header</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description of the channel. It is displayed when editing the channel.</td>
</tr>
<tr>
<td>Image</td>
<td>Select the image to be displayed for the channel</td>
</tr>
<tr>
<td>Node Kinds</td>
<td>Select the node kinds to add to the channel from the list of available node kinds. Move the selected node kinds to or from the Selected Node Kinds list using the left or right arrows. Reorder the node kinds in the selected list using the up or down arrows.</td>
</tr>
</tbody>
</table>

2. Select **Apply** to create the channel or **Cancel** to discard the changes and return to the channels page.

To add a video channel

You can add a video channel that contains the video. To do this:
1. From the Channels page, click **Add Video Channel**. The Add Video Channel page displays the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the name for the channel. The name can only contain alphanumeric, underscore, and period characters.</td>
</tr>
<tr>
<td>Title</td>
<td>Enter the title to be displayed on UI</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description of the channel. It is displayed when editing the channel.</td>
</tr>
<tr>
<td>HTTP URL</td>
<td>Enter the URL for the video source</td>
</tr>
</tbody>
</table>

2. Select **Apply** to save the changes or **Cancel** to discard the changes and navigate back to the channels page.

To add an RSS channel

You can add an RSS channel that contains RSS feeds. To do this:

1. From the Model section of the **Administration** tab, select **Channels**.  
2. Click **Add RSS Channel**. The Add RSS Channel page displays the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter the name for the channel. The name can only contain alphanumeric, underscore, and period characters.</td>
</tr>
<tr>
<td>Title</td>
<td>Enter the title to be displayed on UI</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description of the channel. It is displayed when editing the channel.</td>
</tr>
<tr>
<td>HTTP URL</td>
<td>Enter the URL for the RSS feed</td>
</tr>
<tr>
<td>Render Description</td>
<td>Select this checkbox to render the description</td>
</tr>
</tbody>
</table>

3. Select **Apply** to save the changes or **Cancel** to discard the changes and navigate back to the channels page.

### Exporting data

BMC Atrium Discovery Export enables you to publish data from BMC Atrium Discovery directly into relational databases and into all systems that can accept CSV files. BMC Atrium Discovery data is exported by creating an **exporter**, which is a combination of a mapping set with an adapter configuration. These concepts are explained later in this section.

Export does not delete any BMC Atrium Discovery data from BMC Atrium Discovery, or from the export target.

- Exporters
- Mapping files and sets
Adapters and adapter configurations

To permit users to edit the adapter configuration

Exporters

An exporter is the combination of a mapping set and an adapter configuration.

- The mapping set is a set of mapping files. These allow you to determine which data to extract from BMC Atrium Discovery and how to transform it according to the target data model.
- The adapter configuration is used to determine where to publish the data.

Mapping files and sets

When performing an export, the exporter needs to know which data to read from the BMC Atrium Discovery datastore and where to place it. It does this with mapping files. A mapping file is an XML document that contains the following information:

1. A BMC Atrium Discovery datastore query specifying which data to export
2. Information on how BMC Atrium Discovery Export must restructure (or transform) the query result in order to make it usable by the system being exported to (that is, to comply with the remote system’s schema)

A mapping set is a set of mapping files that are used to carry out the export. In order to perform an export using a mapping set, the exporter will run the query from each one of the mapping files in the set. A typical export requires more than one query of the datastore. For example, one query to get hosts, another to get Business Application Instances (BAIs), and so on.

For further information about mapping file formats, see The Mapping File Format.

Adapters and adapter configurations

BMC Atrium Discovery Export supports the following adapters:

- RDB — populates relational databases
- CSV — generates CSV files

An adapter configuration typically contains the information needed to connect to the target destination:

- Target host
- Port
- User name
- Password
You can create multiple adapter configurations. For example, if you needed to export data from BMC Atrium Discovery to two relational databases ("development" and "production") then you would create two adapter configurations for the RDB adapter, one for each destination.

**To permit users to edit the adapter configuration**

The cmdb-export-administrator user group gives permissions for users to edit the adapter's configuration. To add a user to this group:

1. From the BMC Atrium Discovery appliance UI, log in to BMC Atrium Discovery.
2. Click the **Administration** tab in the top-right corner of the window.
3. Click the **Users** icon in the **Security** channel.
4. Edit the profile of any user that you want to administer BMC Atrium Discovery Export. Add the user to the cmdb-export-administrator group. The user must be a member of the public group.

**Setting up and configuring an export**

This topic provides information about what you will need to do to create and configure an exporter from scratch. Information about managing the adapter configurations, mapping sets and exporters is also provided.

**To set up an export**

1. Specify where the exporter should send data to by creating a new configuration for the appropriate adapter. Configure with the connection settings for the server you want to export to. See [Creating a New Adapter Configuration](#).
2. Specify which data to export by creating a new mapping set and uploading the mapping files you want to use. You can reuse one of the existing mapping sets. See [Creating Mapping Sets](#).
3. Create an exporter, selecting the adapter configuration and mapping set you created in the previous steps. See [Creating a New Exporter](#).

**To start an export**

1. From the Model section of the Administration page, click the **Export** icon. The Integrations page is displayed which has the following tabs:
   - **Exporters** tab
     - Creating a New Exporter
     - Deleting an Exporter
     - Running an export
   - **Adapter Configurations** tab
     - Creating a New Adapter Configuration
1. Editing Adapter Configurations
2. Deleting Adapter Configurations

3. Mapping Sets tab
   - Creating Mapping Sets
   - Editing Mapping Sets
   - Deleting Mapping Sets

⚠️ Export Configurations Backed up by appliance backup

The Appliance Backup feature now backs up export configurations.

Managing adapter configurations
This topic contains information and instructions about managing adapter configuration.

- To create an adapter configuration
- To edit the adapter configurations
- To delete adapter configurations

To view the configuration page for the adapter configurations, click the Adapter Configurations tab. The adapter configurations page contains a section for each adapter:

- CSV file
- Relational database

When the page is first displayed, the sections are collapsed. You can expand them by clicking on the name of the adapter. When the section is expanded, a table listing the configurations for the adapter is visible.

To create an adapter configuration

1. Navigate to the export user interface (UI) by choosing Administration > Export. If an export has never been set up, links display all the available adapter types.
2. Click the appropriate type to create an adapter configuration.
3. If an export has already been created and you want to create another, click the Adapter Configurations tab and then the New link for the corresponding adapter configuration type. A page displays with the details of your new configuration.

The options displayed depend on the type of adapter configuration that is created. A CSV configuration has different options than a relational database (RDB) configuration. However, the following options are common to both:

- Name: The name of an adapter configuration is its identifier in the system. The only characters that can be used in the configuration name are:
  - a to z
  - A to Z
3. Description: The description of an adapter configuration is only for your use. You can enter any information that you want to record about the adapter. For information about the per-adapter options, see:

- Configuring the CSV file adapter for the CSV file configuration.
- Configuring the RDB adapter for the RDB adapter configuration.

4. To save the new adapter configuration, click **Apply Only**.

5. To save the changes and verify that the adapter can connect to the target system with the details supplied, click **Apply and Test** (see below for more details).

   If a validation error occurs on any of the details that you have entered, an error message is displayed.

   If the connection test fails, a page is displayed with a message and a button which enables you to edit the adapter configuration. You cannot click **Back** from this page in order to edit and correct the configuration details. Your adapter configuration has already been created. Clicking **Back**, editing the options and clicking **Apply** results in an attempt to create another configuration with the same name. This causes an error.

6. To edit the adapter configuration that resulted in a failed connection, click **Edit This Adapter**.

   **To edit the adapter configurations**

   1. Click the **Edit** link for an adapter configuration.
      The configuration edit page is displayed.
      The options displayed depend on the type of adapter configuration being edited; each adapter configuration has different options.
      However, the following options are common to both:

      - **Name**: The name of an adapter configuration is its identifier in the system. The name is not modifiable after the configuration has been created.
      - **Description**: The description of an adapter is only for your use. You can enter any information that you want to record regarding the adapter. For information about the per-adapter options, see the following pages:

      - Configuring the CSV file adapter for the CSV file configuration.
      - Configuring the RDB adapter for the RDB adapter configuration.

   2. To save the changes you have made to the options, click **Apply Only**.

   3. To save the changes you have made to the options and verify that the adapter can contact the target system with the details you have supplied, click **Apply and Test**.

   **To delete adapter configurations**

   You can delete a configuration by clicking on the **Delete** link in the Actions field. Be aware that after a configuration is deleted, you cannot retrieve it. Also, if you delete a configuration that is used by exporters, those exporters are also deleted.
Configuring the CSV file adapter

This topic provides instructions for configuring the CSV file adapter.

- To configure a CSV file adapter
- Output files

To configure a CSV file adapter

1. From the Adaptor Configurations tab, select the New link on the CSV File Configurations row.
2. Specify the protocol over which you want to publish the generated CSV files.
   The CSV adapter supports publishing over a variety of file transfer protocols:

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Networking (SMB)</td>
<td>Windows file shares: requires a Microsoft Windows server or a UNIX Samba server sharing a directory with write permissions. <strong>Note:</strong> If you are using a Windows domain account as the credential to access the share then you need to specify it as <code>domain\username</code>.</td>
</tr>
<tr>
<td>SCP</td>
<td>Secure Copy - requires a host running an SSH server. Supports password authentication only.</td>
</tr>
<tr>
<td>SFTP</td>
<td>Secure FTP - requires a host running an SSH server capable of supporting the SFTP protocol. Supports password authentication only.</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol - requires a host running an FTP server.</td>
</tr>
<tr>
<td>Webserver permitting HTTP PUT</td>
<td>Requires a web server hosting a URL that supports the HTTP PUT action. Supports BASIC authentication only.</td>
</tr>
</tbody>
</table>

⚠️ **SMB export over IPv6 not supported on RHEL 5 appliances**

If you upgraded from BMC Atrium 8.3.x rather than migrating, your appliance is still running on RHEL 5 and you cannot export to SMB shares over IPv6. Error rendering macro ‘excerpt-include’ : No link could be created for ‘Cannot scan IPv6 addresses’.

3. If you want to use the default port for the protocol, leave the Specify TCP Port check box unchecked. Otherwise, check it and enter the TCP port.
   The following table lists the default ports for the various protocols. Even though the BMC Atrium Discovery appliance firewall is already configured to allow these connections, you might need to open these ports on your firewall(s) to enable the file publishing.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMB (Windows networking)</td>
<td>TCP 445</td>
</tr>
<tr>
<td>SCP and SFTP</td>
<td>TCP 22</td>
</tr>
<tr>
<td>FTP</td>
<td>TCP 21</td>
</tr>
</tbody>
</table>
3.

4. In the **Username** field, enter the user name BMC Atrium Discovery Export should log in to the remote system as.

5. Enter the password for the specified user into the **Password** field.
   - To set a blank password, leave the **Password** field blank.
     If editing an adapter configuration, select the check box and leave the field blank.
   - To leave the password unchanged, do not select the check box. The password defaults to blank.

6. Specify the path to copy the CSV files to.
   For Windows networking publishing, the path delimiter is \ (backslash), whereas all the other publishers use a / (forward slash). For example:
   - To publish the CSV files to a Windows UNC path of \myserver\myshare\mysubdir, set the **Destination Server** field to myserver (no slashes) and the **Path** field to myshare\mysubdir (note the relative path specified by no leading slash).
   - To publish the CSV files to the directory /tmp/ over SCP, set the **Path** to /tmp (note the absolute path specified by the leading forward slash).
   - To publish the CSV files to the home directory of your user's account over SCP, leave the **Path** blank.
   - To publish the CSV files to a directory beneath your user's home directory over SCP ~\mydir\mysubdir, specify the **Path** as mydir/mysubdir (note that the path is specified as relative to the user's home directory by omitting the leading forward slash).
   - To publish the CSV files over HTTP PUT to the URL http://mywebserver/mypath/mysubpath, specify the **Destination Server** as mywebserver and the **Path** as mypath/mysubpath.

By default, the CSV adapter will output data files that are compatible with Microsoft Excel and other popular spreadsheet applications. Changing the following settings could prevent these products from reading the files successfully.

1. Select the separator character from the drop-down list.
   The available separator characters are comma, semicolon, or tab. The default character is the same as the option set in the **Application Preference** page.

2. Uncheck **Write Header Row** if you do not want each CSV file's column names to be written to the first line of the file.
   By default the CSV adapter will copy a manifest file with the generated CSV files. The manifest records the date of the export and the files that form part of it.
3. Uncheck **Include a Manifest File** to disable the inclusion of the manifest. When you test the connection, BMC Atrium Discovery Export attempts to contact the host, log in by using the credentials provided, and write a test file (called *connection.test* ) to the specified location. If there are problems, an error message is displayed; correct the error.

**Output files**

When generating files, the exporter will generate one CSV file for every CI type in the mappings. Information about CI types can be found in **Understanding the export process**. Each CSV file will be named as follows:

<mapping file name>-<ci name>.csv

For example, if you have a mapping file named *hosts.xml* containing three CIs (host, ip_endpoint, and sc), the following files will be generated:

- **hosts-host.csv**
- **hosts-ip_endpoint.csv**
- **hosts-sc.csv**

**Configuring the RDB adapter**

BMC Atrium Discovery Export includes a relational database (RDB) adapter that allows you to export and publish data to a specific set of database tables over JDBC.

- **Before you begin**
- **To configure the RDB adapter**
- **To write export statistics to a manifest table**
- **To determine the table structure for your mapping files**

**Before you begin**

Prior to creating an RDB adapter configuration, you will need to set up the database where the data will be exported to.

For further information about the database configuration, see **#RDB Adapter - Determining the Table Structure for your Mapping Files**.

**To configure the RDB adapter**

1. From the Administration page, then click the **Export** link and then click the **Adapter Configurations** tab.
2. Click the **New** link next to Relational Database Configurations. Export includes a number of database drivers (for SQL Server, Sybase and PostgreSQL). To export data to other database products you need to provide the appropriate JDBC driver. See **Adding a new JDBC driver for RDB adapter** for information on how to do this.
3. Select the driver from the **Database Driver** drop-down list.
4. Enter the URL to connect to the JDBC database in the Connection URL field. For example, `jdbc:jtds:sqlserver://mydbserver/export`
5. Enter the RDB user name into the **Username** field.

6. Select the **Set Password** check box (only available if you are editing an existing adapter) and enter the password in the **Password** field.

   The JDBC adapter can write export statistics and dates to a manifest table after each mapping is complete. This feature is enabled by default on each adapter configuration. The manifest records the date of the export and the files that form part of it.

7. To disable the inclusion of the manifest, uncheck **Write export statistics to a manifest table**.

**To write export statistics to a manifest table**

The JDBC adapter can write export statistics and dates to a manifest table after each mapping is complete. This feature is enabled by default on each adapter configuration.

If the feature is enabled but the table does not exist in the database, an error will be logged in the export logs after each mapping. The export will continue normally.

**To create the manifest table in the database**, use the following SQL command:

```sql
create table manifest (
    export_item varchar(50),
    insert_count int,
    ignore_count int,
    fail_count int,
    start_date timestamp,
    complete_date timestamp)
```

**To determine the table structure for your mapping files**

The RDB adapter exports data to tables in the database that are determined by the names used in the mapping files. This section describes how to determine the table structure that will be required for your mapping files.

See **The Mapping File Format** for more information about mapping files. For your reference, there is generic SQL at the beginning of each mapping file to delete and create each of the required staging tables for that file.

The RDB adapter exports to a set of staging tables in a manner similar to a CSV export. It requires that the destination tables are empty when the export begins, as it performs no reconciliation and will attempt to **INSERT** records only.

The tables and fields that it exports to are determined by a simple set of rules:

1. Each CI declaration in the mapping file exports to a different table.
2. The table name is formed by concatenating the mapping file's name (without the `.xml` extension), an underscore `_` and the cmdb-name attribute of the CI.
3. The fields of the table exported to by the main CI are taken from the fields declared in the main CI as-is.

4. The fields of the table exported to by all sub CIs are taken from two places:
   - The first fields in the table are the main CI's identity fields.
   - The remaining fields in the table are the sub CI's fields.

Thus, the main CIs are exported to a table of their own. The sub-CIs are exported to a table of their own, but each sub-CI table contains the identity fields of the main CI that it was related to when it was exported.

For example, the following are the CI declarations from `host.xml` file, a mapping file shipped in the RDB mapping set:

```xml
<ci cmdb-name="host" main="true">
    <field src="hostname" dest="host_HostName" identity="true"/>
    <field src="name" dest="host_Name" identity="true"/>
    <field src="description" dest="Description"/>
    <field src="domain" dest="Domain"/>
    <field src="model" dest="Model"/>
    <field src="serial" dest="Serial"/>
    <field src="ram" dest="RAM"/>
    <field src="workgroup" dest="Workgroup"/>
    <field src="vendor" dest="Vendor"/>
    <field src="os" dest="OS"/>
    <field src="os_version" dest="OsVersion"/>
    <field src="processor" dest="Processor"/>
</ci>

<ci cmdb-name="si" collection="true">
    <field src="si_name" dest="SI_Name" identity="true"/>
    <relationship cmdb-name="hostedsoftware" direction="main-to-sub"/>
</ci>

<ci cmdb-name="ipendpoint" collection="true" parent-is-identifier="true">
    <field src="netmask" dest="ipendpoint_Netmask" identity="true"/>
    <field src="ip_addr" dest="ipendpoint_IPAddress" identity="true"/>
    <relationship cmdb-name="deviceinterface" direction="main-to-sub"/>
</ci>
```

There are three CI declarations. The first is the main CI, and the following two are both sub-CIs. Therefore, this mapping file will export data to three tables:

1. host_host - the "host" main CI
2. host_si - the "SI" sub-CI
3. host_ipendpoint - the "ipendpoint" sub-CI
Each table name is formed by concatenating the mapping file name, an underscore and the CI name.

The following is the SQL required for each CI's table. This SQL, while generic, might need slight modifications to run in your database:

```sql
drop table host_host;
create table host_host (  
    host_HostName varchar(100) NOT NULL,  
    host_Name varchar(100) NOT NULL,  
    Description varchar(100),  
    Domain varchar(100),  
    Model varchar(100),  
    Serial varchar(100),  
    RAM integer,  
    Workgroup varchar(100),  
    Vendor varchar(100),  
    OS varchar(100),  
    OsVersion varchar(100),  
    Processor varchar(100),  
    primary key (host_HostName, host_Name)
);
```

The fields for the host_host table were taken directly from the previous host CI declaration.

```sql
drop table host_si;
create table host_si (  
    host_HostName varchar(100) NOT NULL,  
    host_Name varchar(100) NOT NULL,  
    SI_Name varchar(100) NOT NULL,  
    primary key (host_HostName, host_Name, SI_Name)
);
```

The first two fields in the host_si table are the host CI's identity fields, and the remainder are the fields from the SI sub-CI.

```sql
drop table host_ipendpoint;
create table host_ipendpoint (  
    host_HostName varchar(100) NOT NULL,  
    host_Name varchar(100) NOT NULL,  
    ipendpoint_Netmask varchar(100) NOT NULL,  
    ipendpoint_IPAddress varchar(100) NOT NULL,  
    primary key (host_HostName, host_Name, ipendpoint_Netmask, ipendpoint_IPAddress)
);
```
As with the SI sub-CI, the ipendpoint CI’s fields are the host’s identity fields and the ipendpoint fields.

You should create primary keys in the tables that match the identity fields declared in the mapping files. This will prevent the exporter from inserting duplicate rows, or at least highlight that duplicate rows could exist.

**Adding a new JDBC driver for RDB adapter**

A JDBC driver is shipped in the form of a single file with an extension of `.jar` (Java ARchive). The JAR file contains a compiled java implementation of the client code to remotely talk to the database. Database vendors usually make their JDBC drivers available for free download from their websites.

Due to licensing issues we cannot ship the drivers for all of the databases that we support. They are available from the vendors’ websites for download. We supply the URLs to access these jar files on the JDBC Drivers page. To access this, in the Administration page, Appliance section, click **JDBC Drivers**.

After you have downloaded the JDBC driver for the database you wish to communicate with, upload it to the appliance using the correct *Upload* link in the table. The file will be uploaded and verified. If the required Java class is not found in the jar, or if the jar is corrupt (or not a jar at all) then an appropriate message will be displayed in the Status column. Simply re-upload the correct driver to correct the issue should one arise.

<table>
<thead>
<tr>
<th>tideway service restart required</th>
</tr>
</thead>
<tbody>
<tr>
<td>You must restart the tideway services before using the newly uploaded JDBC driver.</td>
</tr>
</tbody>
</table>

The available statuses for the drivers are:

- No JAR Uploaded
- Jar Uploaded (Deactivated)
- Activated and in use. This means the driver is being used by an SQL credential.
- Activated but not in use
- Error.

BMC Atrium Discovery is shipped with a default set of properties files which define the databases that you can connect to. If you want to connect to a different database, you must write a properties file and download an appropriate driver. This is described in **Adding new JDBC drivers**. The following table provides an example JDBC URL for database targets. It also provides documentation and download URLs where available. When selecting the appropriate driver, note that BMC Atrium Discovery uses JDK 1.8. Consult the database vendor’s documentation for further information.
<table>
<thead>
<tr>
<th>Database</th>
<th>JDBC URL</th>
<th>Internet site</th>
</tr>
</thead>
<tbody>
<tr>
<td>Postgres</td>
<td>[jdbc:postgresql://host:[port]/database [?propertyName1]=propertyValue1[&amp;propertyName2]=propertyValue2]</td>
<td>The following example URL connects to a Postgres server running on a host on IP address 192.168.0.100, port 5432, and database name ADDM database. [jdbc:postgresql://192.168.0.100:5432/ADDMdatabase]</td>
</tr>
<tr>
<td>Oracle</td>
<td>For Oracle there are two possible connection styles: <strong>Connection using service</strong> [jdbc:oracle:&lt;drivertype&gt;:[username/password]@//host[:port]/service] The following example URL connects, using a thin driver, to an Oracle server running on a host on IP address 192.168.0.100, port 1521, and service ADDM_DB. [jdbc:oracle:thin:@//192.168.0.100:1521/ADDM_DB] <strong>Connection using SID</strong> [jdbc:oracle:&lt;drivertype&gt;:@host[:port]:SID] The following example URL connects, using a thin driver, to an Oracle server running on a host on IP address 192.168.0.100, port 1521, and SID ADDM100. [jdbc:oracle:thin:[username/password]@192.168.0.100:1521:ADDM100]</td>
<td>Download: <a href="http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html">http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html</a> Documentation: <a href="http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html">http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-faq-090281.html</a> See here for more information on setting up both styles in BMC Atrium Discovery.</td>
</tr>
<tr>
<td>Ingres</td>
<td>[jdbc:ingres://host:[port]/database [;property=value] [;property=value]]</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td>JDBC URL Internet site</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>------------------------</td>
<td></td>
</tr>
<tr>
<td>Ingres</td>
<td>The following example URL connects to an Ingres server running on a host on IP address 192.168.0.100, port mnemonic II7, database name ADDMdatabase, user fred, and password password. jdbc:ingres://192.168.0.100:II7/ADDMdatabase;user=fred;password=password. Download: <a href="http://community.ingres.com/wiki/JDBC_Driver">http://community.ingres.com/wiki/JDBC_Driver</a> Documentation: <a href="http://community.ingres.com/wiki/Open_Office_How_To#Ingres_JDBC_URL_Connection_Information">http://community.ingres.com/wiki/Open_Office_How_To#Ingres_JDBC_URL_Connection_Information</a></td>
<td></td>
</tr>
<tr>
<td>Sybase</td>
<td>The following example URL connects to a Sybase server running on a host on IP address 192.168.0.100, port 6689, and database name ADDM_DB. jdbc:sybase:Tds:192.168.0.100:6689/ADDM_DB Documentation: <a href="http://www.sybase.com/detail?id=1009876#sec2q2">http://www.sybase.com/detail?id=1009876#sec2q2</a></td>
<td></td>
</tr>
<tr>
<td>MS SQL</td>
<td>The following example URL connects to an instance of MS SQL server called TDA running on a host on IP address 192.168.0.100, port 1433, user fred, and password password. The username and password correspond to a user configured on the database rather than a Windows AD user. See this Microsoft article for more information. jdbc:sqlserver://192.168.0.100\TDA:1433;User=fred;Password=password Download: <a href="http://msdn.microsoft.com/en-us/data/aa937724.aspx">http://msdn.microsoft.com/en-us/data/aa937724.aspx</a> Documentation: <a href="http://msdn.microsoft.com/en-us/library/ms378428.aspx">http://msdn.microsoft.com/en-us/library/ms378428.aspx</a> The downloaded file (a gzipped tar archive, current version sqljdbc_3.0.1301.101_enu.tar.gz) contains two JAR files. Use the sqljdbc4.jar the other does not work.</td>
<td></td>
</tr>
<tr>
<td>JTDS</td>
<td>The following example URL connects using JTDS to an MS SQL server server running on a host on IP address 192.168.0.100, port 1433, database name ADDM_IMPORT, instance TDA, user fred, and password password. jdbc:jtds:sqlserver://192.168.0.100:1433/ADDM_IMPORT;instance=TDA;user=fred;password=password When using a domain credential (Windows Authentication) of the form DOMAINNAME\username enter the username in the URL described, and the domain information in the Additional JDBC parameters dialog box in the following form: domain=&quot;DOMAINNAME&quot;. Also, if the domain controller requires NTLM v2 add the parameter: useNTLMv2=true The following example URL connects using JTDS to an MS SQL server server running on a host on IP address 192.168.0.100, port 1433, database name ADDM_IMPORT, instance TDA, Windows user fred in the domain DOM1, and password password.</td>
<td></td>
</tr>
</tbody>
</table>
### Managing mapping sets

This topic provides instructions for managing mapping sets

- To view the page for the mapping sets
- To view or download mapping files
- To create a mapping set
- To edit a mapping set
- To delete a mapping set

#### To view the page for the mapping sets

1. Click **Administration**.
2. Click the **Export in the Model** channel.
3. Click the **Mapping Sets** tab.

The page contains a section for each mapping set. When the page is first displayed the sections are collapsed. You can expand them by clicking on the name of the mapping set. In the screen shot above, the atrium2-mapping-set is expanded, while the others are collapsed.

When a section is expanded, the details of the mapping set are visible. This includes a table of the mapping files that comprise the mapping set.

Export includes the following default mapping sets:

- **ASG (asg-mapping-set)** — See ASG Sample Mapping Set for details.
- **Compuware (compuware-mapping-set)** — See Compuware Sample Mapping Set for details.
- **CSV (csv-mapping-set)** — See CSV Sample Mapping Set for details.
- **Extended RDB (extended-rdb-mapping-set)** — See Extended RDB Sample Mapping Set for details.
- **RDB (rdb-mapping-set)** — see RDB Sample Mapping Set for details.

---

**Database**

<table>
<thead>
<tr>
<th>JDBC URL</th>
<th>Internet site</th>
</tr>
</thead>
<tbody>
<tr>
<td>jdbc:jtds:sqlserver://192.168.0.100:1433/ADDM_IMPORT;instance=TDA;domain=DOM1;useNTLMv2=true;user=fred;password=password</td>
<td></td>
</tr>
</tbody>
</table>

**Documentation:** http://jtds.sourceforge.net/faq.html#urlFormat

---

⚠️ **No IPv6 support**

IPv6 access using JTDS is not currently possible.
To view or download mapping files

To view a mapping file, click its **Download** link. The browser asks you what you want to do with the file. Choose **Open** to view it, or **Save to disk** to download it.

To create a mapping set

1. Click **Create New Mapping Set**.
2. On the Create New Mapping Set page, enter the following details:
   - **Name** — Name of the new mapping set. The name of a mapping set is its identifier in the system. The only characters that you can use in the mapping set name are, `a` to `z`, `A` to `Z`, `0` to `9`) and `-` (dash), `_` (underscore) and `.` (period).
   - **Description** — Free text description of the new mapping set
3. Select one or more mapping files to add to the mapping set. Click **Browse** and use the File upload dialog to locate the mapping files on your file system.
4. Click **Apply** to create the new mapping set.

To edit a mapping set

1. Click the **Edit** link for the mapping set.
   The name is not modifiable after the mapping set is created.
2. To delete mapping files from the set, check the files that you want to delete.
3. To upload new mapping files to the mapping set, use the **Browse** button to select the files to upload.
4. To abandon your changes and return to the **Mapping Sets** tab, click **Cancel**.
5. To save your changes (including deleting any checked files and uploading any added files) and then return to the **Mapping Sets** tab, click **Save**.

If you attempt to upload an invalid mapping file or omit the description, an error is displayed. You must re-add the new mapping files after correcting the error.

To delete a mapping set

To delete a mapping set, click the **Delete** link.

After a mapping set has been deleted, there is no way to retrieve it. If you delete a mapping set that is used by one or more exporters, those exporters are deleted too.

Sample mapping sets

This section provides sample mapping sets you can extend as you need.

- **ASG sample mapping set**
- **Compuware sample mapping set**
- **CSV sample mapping set**
- **Extended RDB sample mapping set**
- **RDB sample mapping set**
ASG sample mapping set

The mapping set used to integrate with the ASG MetaCMDB product for organizations implementing Business Service Management (BSM).

<table>
<thead>
<tr>
<th>File Name</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>bai.xml</td>
<td>Export BAIs and their relationships to Hosts</td>
</tr>
<tr>
<td>si.xml</td>
<td>Export SIs and their BAI relationships</td>
</tr>
<tr>
<td>host.xml</td>
<td>Export Hosts and their SI relationships</td>
</tr>
<tr>
<td>switch.xml</td>
<td>Exports switches and their relationships to Hosts</td>
</tr>
</tbody>
</table>

Compuware sample mapping set

This is a sample mapping set to be used for exporting data to Compuware.

<table>
<thead>
<tr>
<th>File name</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additions_Host.xml</td>
<td>Export New SI-Host Dependencies.</td>
</tr>
<tr>
<td>Additions_NIC.xml</td>
<td>Export New Host-NIC Dependencies.</td>
</tr>
<tr>
<td>Additions_SI.xml</td>
<td>Export New BAI-SC Dependencies.</td>
</tr>
<tr>
<td>Additions_Switch.xml</td>
<td>Export New NIC-Switch Dependencies.</td>
</tr>
<tr>
<td>BAI-SI.xml</td>
<td>Export BAI/SI/BAI-SI Dependencies.</td>
</tr>
<tr>
<td>Changes_Host.xml</td>
<td>Export Host Attribute Changes.</td>
</tr>
<tr>
<td>Changes_NIC.xml</td>
<td>Export NIC Attribute Changes.</td>
</tr>
<tr>
<td>Changes_Switch.xml</td>
<td>Export Switch Attribute Changes.</td>
</tr>
<tr>
<td>Deleted_BAI.xml</td>
<td>Export Destroyed BAIs.</td>
</tr>
<tr>
<td>Deleted_Host.xml</td>
<td>Export Destroyed Hosts.</td>
</tr>
<tr>
<td>Deleted_NIC.xml</td>
<td>Export Destroyed NICs.</td>
</tr>
<tr>
<td>Deleted_SI.xml</td>
<td>Export Destroyed SIs.</td>
</tr>
<tr>
<td>Deleted_Switch.xml</td>
<td>Export Destroyed Switches.</td>
</tr>
<tr>
<td>Host-NIC.xml</td>
<td>Export NICs/Host-NIC Dependencies.</td>
</tr>
<tr>
<td>Host.xml</td>
<td>Export Hosts.</td>
</tr>
<tr>
<td>Nic-Switch.xml</td>
<td>Export NIC-Switch Dependencies.</td>
</tr>
<tr>
<td>SI-Host.xml</td>
<td>Export SI-Host Dependencies.</td>
</tr>
<tr>
<td>Switch.xml</td>
<td>Export Switches.</td>
</tr>
</tbody>
</table>

For further information about the Compuware mapping set see The Compuware mapping set.
CSV sample mapping set

The default mapping set provided for use with the CSV adapter uses the following mapping files to export the nodes listed below:

<table>
<thead>
<tr>
<th>File name</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>bai.xml</td>
<td>Export BAIs and their relationships to Hosts.</td>
</tr>
<tr>
<td>si.xml</td>
<td>Export SIs and their BAI relationships.</td>
</tr>
<tr>
<td>host.xml</td>
<td>Export Hosts and their SI relationships.</td>
</tr>
<tr>
<td>switch.xml</td>
<td>Exports switches and their relationships to Hosts.</td>
</tr>
</tbody>
</table>

Extended RDB sample mapping set

This is a sample mapping set which can be used to perform a more complete export of BMC Atrium Discovery data to the RDB.

<table>
<thead>
<tr>
<th>File name</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>package.xml</td>
<td>Export Packages and their Hosts relationships.</td>
</tr>
<tr>
<td>host.xml</td>
<td>Export Hosts and their SI relationships.</td>
</tr>
<tr>
<td>file.xml</td>
<td>Export Files and their Host/SI relationships.</td>
</tr>
<tr>
<td>cluster.xml</td>
<td>Export Clusters and their Hosts relationships.</td>
</tr>
<tr>
<td>bai.xml</td>
<td>Export BAIs and their relationships to Hosts.</td>
</tr>
<tr>
<td>virtualcontainer.xml</td>
<td>Export Virtualisation and their Host relationships.</td>
</tr>
<tr>
<td>switch.xml</td>
<td>Exports switches and their relationships to hosts.</td>
</tr>
<tr>
<td>si.xml</td>
<td>Export SIs and their relationships.</td>
</tr>
<tr>
<td>patch.xml</td>
<td>Export Patches and their Hosts relationships.</td>
</tr>
</tbody>
</table>

For further information about the extended RDB mapping set see The extended RDB mapping set.

RDB sample mapping set

The default mapping set provided for use with the RDB adapter uses the following mapping files to export the nodes listed below:

<table>
<thead>
<tr>
<th>File name</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>bai.xml</td>
<td>Export BAIs and their relationships to Hosts.</td>
</tr>
<tr>
<td>si.xml</td>
<td>Export SIs and their BAI relationships.</td>
</tr>
<tr>
<td>host.xml</td>
<td>Export Hosts and their SI relationships.</td>
</tr>
<tr>
<td>switch.xml</td>
<td>Exports switches and their relationships to Hosts.</td>
</tr>
</tbody>
</table>

For further information about the extended RDB mapping set see The extended RDB mapping set.
Managing exporters

This topic provides instructions for managing exporters.

- To view the configuration page for the exporters
- To create a new exporter
- To delete an exporter

To view the configuration page for the exporters

1. Click Administration.
2. From the model section, select Export.
   The Exporters tab is displayed by default.

The page contains a section for each configured exporter. No exporters are configured by default. When the page is first displayed the sections are collapsed. You can expand them by clicking on the name of the exporter.

If the Create New Exporter button is not visible, this means that you have either no Adapter Configurations or Mapping Sets installed on your appliance.

When the section is expanded, the details of the exporter, the adapter configuration and mapping set are visible.

To create a new exporter

1. Click the Create New Exporter button.
   A page will be displayed for you to enter the details of the new exporter.
   - **Name** — Name of the new exporter. The name of an exporter is its identifier in the system. The only characters that you can use in the exporter name are, a to z, A to Z, 0 to 9 and - (dash), _ (underscore) and . (period).
   - **Description** — Free text description of the new mapping set.
2. Select the Mapping Set to use for this exporter.
3. Select the Adapter Configuration to use for this exporter.
4. If you want this exporter to always export all the data returned by its queries, uncheck Export Changed Items Only. By default the exporter only exports items that are new or have changed since the last export.
5. To abandon your exporter and return to the Exporters tab, click Cancel.
6. To save the new exporter, click Save.
   An error message reports any problems with the data you have entered.

To delete an exporter

To delete an exporter, click its Delete link.

After an exporter has been deleted, there is no way to retrieve it. Deleting an exporter does not delete the mapping set or adapter configuration.
Running an export

This topic provide instructions for running an export.

- To run an export from the UI
- To run an export from the command line
- To export log files
- Other log files

⚠️ Warning

Only run exports from BMC Atrium Discovery during time periods when no discovery runs are occurring. This ensures that the state of the datastore does not change during the export. Such a change could result in inconsistent data being exported.

There are two ways to run an export:

- From the user interface
- From the command line

To run an export from the UI

1. Navigate to Exporters tab.
2. Click Run for the exporter.
   - Only one export can be run at a time. As a result all of the other exporters' run links become unavailable when an export is running.
   - The output for the currently executing exporter displays beneath the configuration tabs. It refreshes every few seconds, keeping the output up to date.
3. After the export has finished, the output window displays No export is running and stops refreshing.
4. To re-enable the other exporters’ Run links, close the output window.

To run an export from the command line

1. Log in to the appliance as the Tideway user.
2. Run the following command (where exporter-name is the name of the exporter you want to run):
   
   ```bash
   $TIDEWAY/java/integrations/bin/export.sh exporter-name
   ```
   The export now runs and the clean log (see Log Files) is written to the console. Future exports can be scheduled by using a cron job.

To export log files

There are two types of log files produced by Export.
• **Export Log Files**

• **Other Log Files**

Each export writes logs to the normal BMC Atrium Discovery appliance log location. To view the logs, select **Appliance > View Logs**.

The following logs are written:

• **Clean log** — Contains information on what the exporter did with each item. These log files are called `tw_cmdb-export.timestamp.log` where `timestamp` is the date and time that the export commenced.

• **Full log** — Contains much more verbose information about the export process. This log is intended for debugging. The full log is called `tw_cmdb-export.log`.

When the full log reaches 50 MB, it is rolled. That is, a .1 log number suffix is appended to its filename and existing logs have their log number suffixes incremented. The oldest log is deleted. Ten logs are retained in addition to the current log.

See the **Troubleshooting section** for more information on viewing BMC Atrium Discovery log data.

### Other log files

Several other log files are also generated in the BMC Atrium Discovery Appliance log location directory.

• **Startup log** — When BMC Atrium Discovery is started, a component of BMC Atrium Discovery Export runs to perform some cleanup exercises. The log for this activity is in `export-creds.log`.

• **Connection test log** — Whenever a user tests an adapter configuration connection, the messages for the test are written to a log called `tw_exporter_connection_test.log`.

### Understanding the export process

This section provides a detailed description of the export process and the format and function of mapping files.

• **Export process overview**

• **Key steps**

• **Mapping file format**

• To transform a BMC Atrium Discovery dataset by using a mapping file

• **A closer look at mapping files**

• **Errors during the mapping validation phase**

• **The mapping file XSD**
Export process overview

When performing an export, the exporter needs to read data from BMC Atrium Discovery's datastore, restructure it so that it matches the schema of the system it is being exported to and export it to the remote system. If any errors occur during the export of the data then the exporter needs to decide how much of the data to roll back.

To accomplish this, the exporter goes through the following steps:

1. Determines which system to send the data to, and with which connection parameters. This information comes from the adapter configuration specified by the user.
2. Runs a connection test to the specified remote system.
3. Determines which data is to be exported to the remote system, and how that data is to be restructured during the export.
   This information comes from the mapping set specified by the user. Each mapping file in the mapping set describes a dataset in BMC Atrium Discovery's datastore. For each mapping file, the specified data is retrieved, restructured and exported.
4. After all the mapping files have been run, performs any final tasks (logs the export statistics, writes any required manifests) and closes the connection.

The most complicated part of the export process is the restructuring of data. Each record in the BMC Atrium Discovery dataset specified in the mapping file is converted by the exporter into a set of configuration items (CIs). One CI is conceptually similar to a record in a table. A set of CIs can be thought of as a set of records in various tables that are linked by foreign keys. For example, a set of CIs could contain one host CI, four IPAddress CIs and a CPU CI.

Each set of CIs has one Main CI. All of the others are sub CIs. In the above example, the Host would be the main CI, and the IPAddress and CPU items would be sub CIs.

During the export, each record of the BMC Atrium Discovery data set is converted into one set of CIs. Each set of CIs is exported together. If the export of any CI in the set fails, then the whole set is rolled back. For example, if the CPU CI in our example cannot be inserted because a required field is left blank, then the Host and IP Address items will not be inserted either.

An overview of the process for retrieving, restructuring and exporting the data for each mapping file is shown in the figure below.
There are four key steps in the process:

1. The query string is run. You can find more information about the Query section in a mapping file in the **The Mapping File Format**.
2. The result of running that query is produced.
3. The records are transformed into a set of CIs. This is described in more detail in **Transforming a BMC Atrium Discovery Dataset using a Mapping File**.
4. The set of CIs are inserted into the database.

**Export is not synchronization**

When BMC Atrium Discovery data is exported, the appliance does not delete any previously exported data on the target system. For example, when exporting to an RDBMS, you need to perform a manual table truncate procedure to remove the data.
Mapping file format

This section provides an introduction to the mapping file format. It corresponds to step 1 in the Key Steps diagram above. For further details about the sections of the mapping files see A Closer look at Mapping Files.

Each mapping file is made up of two sections, the query section and the transformation section.

- The query is a standard search service query interpreted by BMC Atrium Discovery. BMC Atrium Discovery uses the query to retrieve information from the datastore and then returns the result to the exporter. Further information about search service can be found in the Using the Search and Reporting service.
- The transformation section specifies how the results of the query will be transformed into the appropriate format for publication by an adapter, for example, CSV files.

The following diagram shows the Query and Transform sections of a default mapping file. The diagram also shows the way the transformation section is divided into Main and Sub CIs.

To transform a BMC Atrium Discovery dataset by using a mapping file

This section describes step 3 in the Key Steps diagram above.

A mapping file contains a BMC Atrium Discovery datastore query. When the mapping file is run, the query is executed on the datastore and the query result from this is used as the source data to the transformation specified in the mapping file.

Consider the following query:

```xml
<xs:complexType name="EAI to BMC_Applications via the direct relationship"></xs:complexType>
<xs:query>
  <!--QML: BusinessApplicationInstance  
  where parseTime("$lastExportFinished") < modified() 
  show name, description, # as nodeRef, 
  #RunningSoftware.HostedSoftware.Host.Host.@ as host_nodeRef, 
  #RunningSoftware.HostedSoftware.Host.Host.hostname as hostname, 
  #RunningSoftware.HostedSoftware.Host.Host.name as host_name

  <collection name="EAI_item">
    <field name="nodeRef" collection="true" node-reference-field="nodeRef" />
    <field name="hostname" collection="true" node-reference-field="hostname" />
    <dependency direction="main-to-sub" collection="true" node-reference-field="host_nodeRef" />
  </collection>
</xs:query>
```
#RunningSoftware:HostedSoftware:Host:Host.hostname as host_hostname,
#RunningSoftware:HostedSoftware:Host:Host.local_fqdn as host_fqdn

This query returns the following result set:

<table>
<thead>
<tr>
<th>name</th>
<th>description</th>
<th>host_hostname</th>
<th>host_fqdn</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Payroll</td>
<td>The payroll application</td>
<td>webserv01</td>
<td>Webserv01.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>london_orcl</td>
<td>London_orcl.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sap_01</td>
<td>Sap_01.mycompany.com</td>
</tr>
<tr>
<td>2 Website</td>
<td>Our company website</td>
<td>Webserv01</td>
<td>Webserv01.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Webserv02</td>
<td>Webserv02.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Webserv03</td>
<td>Webserv03.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>London_orcl</td>
<td>London_orcl.mycompany.com</td>
</tr>
<tr>
<td>3 Employee Expenses</td>
<td>The employee expenses application</td>
<td>webserv01</td>
<td>Webserv01.mycompany.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td>London_orcl</td>
<td>London_orcl.mycompany.com</td>
</tr>
</tbody>
</table>

The first two fields (Name and Description) have returned one value per record. The next two fields, on the other hand, are the result of key expression traversals over a relationship. They each return a sequence of values: one value per relationship that was traversed. They are the result of traversing all relationships of the type RunningSoftware:HostedSoftware:Host from the Business Application Instance (BAI). (There is no need to use explode to cause the key expressions to be treated as separate rows in the output.)

The first BAI (Payroll) had three such relationships, and so the host_hostname and host_fqdn fields returned three values each for that BAI's record. The second BAI (Website) had four such relationships, while the last BAI (Employee Expenses) had two.

Both of the fields that returned sequences (host_hostname and host_fqdn) returned sequences that correspond. The first entry in the Payroll's host_hostname field (webserv01) corresponds to the first entry in Payroll's host_fqdn field. The second and third entries in each field also match.

Using these corresponding sequences, we can compile a list of Hosts that are related to each application. In our example, the Payroll application could be described as follows:

```
Name: Payroll
Description: The payroll application
Hosts:
  Host 1
  Hostname: webserv01
  FQDN: webserv01.mycompany.com
```
Host 2
Hostname: London_orcl
FQDN: london_orcl.mycompany.com

Host 3
Hostname: sap_01
FQDN: sap_01.mycompany.com

We have taken one record from the result set and pivoted it, generating a Business Application Instance CI and three Host CIs from the record. This is how the transformation process works. Consider the following CI declarations from a mapping file (this is described in more detail in A Closer look at Mapping Files).

```xml
<ci cmdb-name="bai" main="true">
    <field src="name" dest="Name" identity="true"/>
    <field src="description" dest="Description"/>
</ci>

<ci cmdb-name="host" collection="true">
    <field src="host_hostname" dest="HostHostName" identity="true"/>
    <field src="host_fqdn" dest="HostFQDN" identity="true"/>
    <relationship cmdb-name="hostedsoftware" direction="main-to-sub"/>
</ci>
```

The first CI (the one declared "main") is the principal CI that this mapping file is concerned with. It is typically the node from which the various traversals start.

The sub-CI ("host") is generated from other fields in the result set. If its fields return sequences then you will need to set "collection='true'"; if you only expect one value per field then you can leave that declaration out.

The "relationship" element in the sub-CI tells the exporter how your main CI and sub CI are related. It is used when exporting to systems where the relationship has a name, such as Atrium CMDB. For the simpler adapters (such as CSV and RDB) it is ignored. If you intend to use the mapping file for these adapters only, you still need to specify the relationship, its name and direction but you can specify any values.

In order for the Exporter to validate mapping files, at least one field in each CI must be given the attribute "identity='true'".

A closer look at mapping files

Query section and the use of timestamp

A sample of the Query section of the mapping file is shown below:
The Query section is built up of search service functions. For more information on how to build search queries, see the Using the Search and Reporting service.

⚠️ **Note**

The following search service functions are not supported by BMC Atrium Discovery Export:

- `dq`
- `dq_band`
- `dq_metric`

In the query section, the exporter makes a variable available that contains the time at which the exporter was last run. This variable is called "lastExportFinished" and is used with the function `parseTime` as follows:

```plaintext
parseTime("${lastExportFinished}")
```

This generates a timestamp that the datastore can recognize.

When this variable is encountered, the exporter substitutes the variable with the date that it was last run. The exporter then sends the search query to the datastore.

By unchecking the "Export changed items only" check box, the exporter will set the `lastExportFinished` to 1 Jan 1980. This will result in a full export.

Example: Using the variable as part of a `where` clause
This variable can be used as part of a `where` clause. The following example will return items that have changed since the last time this exporter was run:

```plaintext
search Host
  where parseTime("${lastExportFinished}") < modified(#)
  show hostname
```
This variable can also be used with search services functions inside mapping file queries. For example, it can be used to filter on changes in dependencies between BAI and software collection.

If you have other conditions to place in the query's *where* clause, it is generally best to put the other conditions before the modified check, to avoid comparing modified times of many nodes that do not match the condition. For example:

```
search Host
where os_type = "Windows" and
parseTime("${lastExportFinished}") < modified(#)
show hostname
```

**Transformation section**

The transformation section is made up of a number of CIs. Each CI has a name (cmdb-name) and a number of field elements. There is one main CI and zero or more sub CIs. There can only be one main element (it has the attribute "main" set to true).

**Main CI transformation**

In this section of the mapping file, the main attribute is set to "true", indicating that this is the main CI.

```xml
<ci cmdb-name="BMC_Application" main="true">
  <field src="name" dest="Name" identity="true"/>
  <field src="description" dest="shortDescription"/>
</ci>
```

The name of the CI on the remote computer is BMC_Application.

The set of fields with `identity = 'true'` together uniquely identify this CI. Identity tags can be set on one or more fields.

⚠️ **Note**

If more than one field has "identity='true'" set then the exporter will only overwrite an existing item if it has identical values in all of the identity fields. In other words, multiple identity fields cause an AND operation, not an OR.
Errors during the mapping validation phase

Errors might be raised during the mapping validation phase. The following table describes these possible errors.

<table>
<thead>
<tr>
<th>Error</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A field cannot have both a 'const' and a 'src' attribute. Field: X</td>
<td>Field X has both a &quot;const&quot; and a &quot;src&quot; attribute specified. These attributes specify where the data for the field will come from; only use one of them.</td>
</tr>
<tr>
<td>Every field must have either a 'const' or a 'src' attribute. Field: X</td>
<td>The specified field did not have either a &quot;src&quot; or &quot;const&quot; attribute. These attributes specify where the data for the field will come from; one of them is required.</td>
</tr>
<tr>
<td>The sub CI X has no relationship configured.</td>
<td>A sub-CI (ie. one without a &quot;main='true'&quot; attribute) needs to have a &quot;Relationship&quot; element. This relationship element is ignored for simple adapters such as the RDB or CSV adapters; it can be specified as:&lt;relationship cmdb-name=&quot;ignored&quot; direction=&quot;main-to-sub&quot;/&gt;</td>
</tr>
<tr>
<td>The CI has no identity fields and no node reference field.</td>
<td>All CIs need to have at least one field marked with &quot;identity='true'&quot;. The exporter uses these attributes to reconcile CIs with those in the destination system. For the RDB and CSV adapters the &quot;identity&quot; attributes on sub-CIs are ignored, but at least one must still be provided — just specify the first field in the CI as &quot;identity='true'&quot;.</td>
</tr>
<tr>
<td>The first field of a collection CI cannot be const.</td>
<td>The first field of a CI with &quot;collection='true'&quot; cannot use &quot;const&quot; as a source. If you only need one data field in your CI, and it is const then export at least one other field of data with the CI.</td>
</tr>
<tr>
<td>The main CI cannot be a collection.</td>
<td>The main CI (ie. the CI with &quot;main='true'&quot;) cannot be a collection CI. Remove the &quot;collection='true'&quot; attribute.</td>
</tr>
<tr>
<td>The main CI cannot have a relationship configured.</td>
<td>The main CI (ie. the CI with &quot;main='true'&quot;) is related to all the other CIs in the set by the relationships configured with those CIs. It cannot have a relationship of its own configured. Remove the &quot;Relationship&quot; element.</td>
</tr>
</tbody>
</table>

The mapping file XSD

Click here to show an example of the XSD that describes the format of the mapping files.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!-- The direction of a relationship between the main and a sub CI -->
  <xs:simpleType name="relationshipDirection">
    <xs:restriction base="xs:string">
      <xs:enumeration value="main-to-sub"/>
      <xs:enumeration value="sub-to-main"/>
    </xs:restriction>
  </xs:simpleType>

  <!-- A field in the CMDB, ie. a CI attribute. -->
  <xs:complexType name="Field">
    <!--...-->
</xs:schema>
```
Set a field's "const" attribute to specify a set value to write to a field in the CMDB. A Field can have either "const" or "src" set, not both.

```xml
<xs:attribute name="const" type="xs:string"/>
```

The field in the query's result set to take this field's value from. You can use aliases for long fields.

For example, use "name" as the src attribute in the following query:

```xml
query host show name
```

Or use the alias name "ip_address" as the src attribute in the following:

```xml
query host show
#HostWithAddress:HostAddress:AddressOfHost:
IpAddress.address as ip_address
```

(The traversal is on two lines for readability, in reality it is on a single line.)

```xml
<xs:attribute name="src" type="xs:string"/>
```

The CMDB class' attribute to write the value to.

```xml
<xs:attribute name="dest" type="xs:string" use="required"/>
```

If this field is required in the CMDB, set this value to "true". If you set this to "true" and a null value is returned from the data store, the sub-CI will be ignored, that is, the exporter will not attempt to insert it into the CMDB.

Failure to do this could cause the CMDB to throw an exception when committing the transaction for the main CI. You would lose the main CI and all of its sub-CIs instead of just the sub-CI with the null field.

```xml
<xs:attribute name="required" type="xs:boolean" default="false"/>
```

Does this field form part of the CI's identity? "false" by default. If any field is set to true then the transform engine will do a lookup in this CMDB class to see if an object with the same values for the identity fields exists, and use that if one is found. If none is found, this item will be inserted. If no fields are marked as identity fields, all items will be inserted as new, even if they're identical.

```xml
<xs:attribute name="identity" type="xs:boolean" default="false"/>
```

```xml
</xs:complexType>
```
<!-- A relationship between the main CI and a sub-CI. -->
<xs:complexType name="Relationship">
  <xs:sequence>
    <!--
    A relationship can have fields in the same way as a CI can.
    -->
    <xs:element name="field" type="Field" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
  <!--
  The cmdb class name for this CI.
  -->
  <xs:attribute name="cmdb-name" type="xs:string" use="required"/>
  <!--
  The direction the relationship goes in. For example, if this is
  the dependency relationship from an application to a host, then
  the application is dependent on the host. The direction is thus
  "main-to-sub" if the main CI is the application.
  -->
  <xs:attribute name="direction" type="relationshipDirection"
    use="required"/>
</xs:complexType>

<!-- A mapping for one CI class -->
<xs:complexType name="CI">
  <xs:sequence>
    <!--
    The fields for this CI
    -->
    <xs:element name="field" type="Field" minOccurs="1"
      maxOccurs="unbounded"/>

    <!--
    How this CI relates to the main CI
    -->
    <xs:element name="relationship" type="Relationship"
      minOccurs="0" maxOccurs="1"/>
  </xs:sequence>

    <!--
    CI class name in the CMDB
    -->
    <xs:attribute name="cmdb-name" type="xs:string" use="required"/>

    <!--
    Is this the "main" CI? There is one main CI and (optionally)
    many sub-CIs.
    -->
    <xs:attribute name="main" type="xs:boolean" default="false"/>

    <!--
    If true (the default), any existing item found in the CMDB based
    on the identity fields will have its non-identity fields
overwritten by the fields from the imported item. If false, the existing object will be left untouched. This value should possibly be set per-attribute.

```xml
<xs:attribute name="overwrite-non-id-fields" type="xs:boolean"
             default="true"/>
```

```xml
<!--
If true, then all of the fields that make up this CI have to be collection fields, and all the collections have to have the same length. The transform engine will generate one sub-CI per set of values in the collections.

If this is false and any of the fields for this CI are returned as a collection, an error will occur.

This is only applicable to sub-CIs. Defaults to "false".
-->
<xs:attribute name="collection" type="xs:boolean"
             default="false"/>
```

```xml
<!--
Some items exist only in the context of their main CIs. A good example is an IP address - the same IP address may exist many times on the network, but will only exist once per Host. Thus, the IP address is identified by its address and its relationship to its main CI, namely the Host.

Set this attribute to true to tell the lookup to treat the relationship to the main CI as part of the identity.
-->
<xs:attribute name="parent-is-identifier" type="xs:boolean"
             default="false"/>
```

```xml
<!--
Specify which field contains a reference to the BMC Atrium Discovery node that this CI is an export of. The exporter uses this reference to store and retrieve the ID given to this CI in the remote system. This ID is then used to reconcile the CI against the remote system before resorting to lookups based on identity fields.

In BMC Atrium Discovery QL, you specify a node reference by "#". For example, you retrieve some fields and the node reference from a Host by:

"search Host show name, fqdn, # as host_ref"

In this example, you would set the "node-reference-field" to "host_ref".
-->
<xs:attribute name="node-reference-field" type="xs:string"/>
```

</xs:complexType>
The Compuware mapping set

This topic provides information about the Compuware mapping set.

- Mapping files
- Configuration on the BMC Atrium Discovery side
- Configuration on the Compuware VSM side

Mapping files

The Compuware mapping set is made up of the following mapping files:

<table>
<thead>
<tr>
<th>File Name</th>
<th>Relationship</th>
<th>TWF nodes and attributes exported</th>
<th>Output CSV Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additions_Host.xml</td>
<td>Export New SI-Host Dependencies.</td>
<td>SI_nodeID, Host_nodeID</td>
<td>Additions_Host-14-TW_Additions_Host.csv</td>
</tr>
<tr>
<td>Additions_NIC.xml</td>
<td>Export New Host-NIC Dependencies.</td>
<td>Host_nodeID, NIC_nodeID</td>
<td>Additions_NIC-15-TW_Additions_NIC.csv</td>
</tr>
<tr>
<td>Additions_SI.xml</td>
<td>Export New BAI-SI Dependencies.</td>
<td>BAI_nodeID, SI_nodeID</td>
<td>Additions_SI-13-TW_Additions_SI.csv</td>
</tr>
<tr>
<td>Additions_Switch.xml</td>
<td>Export New NIC-Switch Dependencies.</td>
<td>NIC_nodeID, Switch_nodeID</td>
<td>Additions_Switch-16-TW_Additions_Switch.csv</td>
</tr>
<tr>
<td>BAI-SI.xml</td>
<td>Export BAI/SI /BAI-SI Dependencies.</td>
<td>• BAI_nodeID, Application, Location, ExportDateFriendly, ExportDate, bai_LastModifiedFriendly, bai_LastModified • SI_nodeID, si_type, si_name, si_version, ExportDateFriendly, ExportDate, si_LastModified, TW_ContainedSI</td>
<td>• BAI-SI-01_TW_BAI_Node.csv • BAI-SI-02_TW_SI_Node.csv</td>
</tr>
<tr>
<td>File Name</td>
<td>Relationship</td>
<td>TWF nodes and attributes exported</td>
<td>Output CSV Files</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Changes_Host.xml</td>
<td>Export Host Attribute Changes.</td>
<td>Host_nodeID, Event Time, Attribute, Value Before, Value After</td>
<td>Changes_Host-10-TW_Changes_Host.csv</td>
</tr>
<tr>
<td>Changes_NIC.xml</td>
<td>Export NIC Attribute Changes.</td>
<td>NIC_nodeID, Event Time, Attribute, Value Before, Value After</td>
<td>Changes_NIC-11-TW_Changes_NIC.csv</td>
</tr>
<tr>
<td>Changes_Switch.xml</td>
<td>Export Switch Attribute Changes.</td>
<td>Switch_nodeID, Event Date, Attribute, Value Before, Value After</td>
<td>Changes_Switch-12-TW_Changes_Switch.csv</td>
</tr>
<tr>
<td>Deleted_BAI.xml</td>
<td>Export Destroyed BAI.s.</td>
<td>BAI_nodeID, Parent</td>
<td>Deleted_BAI-17-TW_Deleted_BAI.csv</td>
</tr>
<tr>
<td>Deleted_Host.xml</td>
<td>Export Destroyed Hosts.</td>
<td>SI_nodeID, Host_nodeID</td>
<td>Deleted_Host-TW_Deleted_Host.csv</td>
</tr>
<tr>
<td>Deleted_NIC.xml</td>
<td>Export Destroyed NIC.s.</td>
<td>Host_nodeID, NIC_nodeID</td>
<td>Deleted_NIC-20-TW_Deleted_NIC.csv</td>
</tr>
<tr>
<td>Deleted_SI.xml</td>
<td>Export Destroyed SIs.</td>
<td>BAI_nodeID, SI_nodeID</td>
<td>Deleted_SI-18-TW_Deleted_SI.csv</td>
</tr>
<tr>
<td>Deleted_Switch.xml</td>
<td>Export Destroyed Switches.</td>
<td>NIC_nodeID, Switch_nodeID</td>
<td>Deleted_Switch-21-TW_Deleted_Switch.csv</td>
</tr>
<tr>
<td>Host-NIC.xml</td>
<td>Export NICs /Host-NIC Dependencies.</td>
<td>• Host_nodeID, NIC_nodeID, ExportDateFriendly, ExportDate, nic_LastModified</td>
<td>• Host-NIC-08-TW_Dependencies_HOST-NIC.csv</td>
</tr>
<tr>
<td>Host.xml</td>
<td>Export Hosts.</td>
<td>Host_nodeID, hostname, fqdn, vendor, model, serial, ram, os, processor</td>
<td>Host-03-TW_Host_Node.csv</td>
</tr>
<tr>
<td>Nic-Switch.xml</td>
<td>Export NIC-Switch Dependencies.</td>
<td>NIC_nodeID, Switch_nodeID, ExportDateFriendly, ExportDate, host_LastModifiedFriendly, host_LastModified</td>
<td>Nic-Switch-09-TW_Dependencies_NIC-Switch.csv</td>
</tr>
<tr>
<td>SI-Host.xml</td>
<td>Export SI-Host Dependencies.</td>
<td>SI_nodeID, Host_nodeID, ExportDateFriendly, ExportDate, si_LastModified</td>
<td>SI-Host-07-TW_Dependencies_SI.csv</td>
</tr>
<tr>
<td>Switch.xml</td>
<td></td>
<td></td>
<td>Switch-05-TW_Switch_Node.csv</td>
</tr>
<tr>
<td>File Name</td>
<td>Relationship</td>
<td>TWF nodes and attributes exported</td>
<td>Output CSV Files</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------</td>
<td>-----------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Export Switches.</td>
<td>Switch_nodeID, SwitchName, SwitchDescription, SwitchModel, SwitchOS_Type, SwitchOS_Version, ExportDateFriendly, ExportDate, switch_LastModifiedFriendly, switch_LastModified</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The integration exports dependency maps to Compuware BSM tool, including information about SW, HW and dependency changes.

**Configuration on the BMC Atrium Discovery side**

The Compuware integration uses the BMC Atrium Discovery CSV Files adapter combined with the Compuware mapping set 7.0. Since Compuware VSM can run on either a Windows or Linux server, the publishing protocol should be chosen accordingly.

The target location for the CSV file adapter specific configuration should be:  
<VSM_INSTALL_DIR>/config/tideway/CSVExports/  
Ask your Compuware VSM admin for the VSM_INSTALL_DIR value.

**Configuration on the Compuware VSM side**

This integration requires an adapter on Compuware side. This is provided by Compuware. The integration uses a Compuware CSV upload adapter which processes all BMC Atrium Discovery generated CSV files imported to the following share:  
<VSM_INSTALL_DIR>/config/tideway/CSVExports/  

**The extended RDB mapping set**

The Extended RDB mapping set provides a set of mapping files for exporting the main BMC Atrium Discovery inferred nodes, except for Host Based Adapters (HBAs).

- Mapping files
- Extended RDB mapping schema
- Configuration on the RDB side

⚠️ **Note**

In version 10.2, the host.xml mapping file was modified: the interface_name, interface_speed, interface_duplex, and interface_negotiation are no longer exported with the Network Interface CI information by default.

If you need to export the mentioned details, customize the Extended RDB adapter mapping set by uploading these host.xml and networkinterface.xml files (for more information on customizing a mapping set, see Managing mapping sets).
When importing data to the external database, you need to modify the default database scripts used for initial configuration on the RDB side to match the introduced changes in the export schema. You can download a sample customized scripts for Oracle, MySQL, and for SQLServer: SQLServer.sql and SQLServer.osql.

**Recommendation**

This mapping set was developed to serve as the basis of a best practice, used to export BMC Atrium Discovery data to RDBMS. It is not intended to be the only best practice, nor does it cover all use cases.

**Mapping files**

The extended RDB Mapping set is designed in such a way that it can be used to form a modular export. That is, the user can pick and match any of the mapping files in the mapping set to decide what they want to export. Each mapping file contains a main node and relevant relationships to other nodes. The exception to this (no relationships) is the Host mapping, as this is expected to form the basis of any export set.

As an example, the BAI mapping file exports to a BAI table, as well as to a BAI-to-Host relationships table. If you include only the Host and BAI mapping files then they will export all data related to both nodes and the relationship links. If you want to include SI in your exports, then this will export the SIs and relationships to BAIs and Hosts.

The idea behind this is to allow flexibility in the export schema while maintaining consistency. If a new node is to be added to the export, then the same process should be applied.

All mapping files will only export data that has been modified since the last export date. This functionality can be overridden by de-selecting the Export Changed Items flag when defining the Exporter.

The Extended RDB mapping set is made of the mapping files as described below.

<table>
<thead>
<tr>
<th>Mapping File</th>
<th>Relationship</th>
<th>Fields</th>
<th>Output Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>bai.xml</td>
<td>BAIs, relationship to Hosts, Cyclic dependencies to: Contained BAIs, DependedOn BAIs</td>
<td>BAI: bai_key, name, description, type, version Host: key Cyclic Dependency BAIs: related BAIKey</td>
<td>bai_ci, bai_contained_bai_rel, bai_dependedon_bai_rel, bai_host_rel</td>
</tr>
<tr>
<td>host.xml</td>
<td>Hosts, Network Interfaces and related Subnet</td>
<td>Host: host_key, host_hostname, host_name, description, domain, model, serial, ram, max_ram, workgroup, vendor, os, os_edition, os_version, os_eos, os_eoes, num_processors, num_logical_processors, processor_type, processor_speed, cores_per_processor</td>
<td>host_ci, host_networkinterface_ci</td>
</tr>
<tr>
<td>Mapping File</td>
<td>Relationship</td>
<td>Fields</td>
<td>Output Table</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>switch.xml</td>
<td>Switches, Switch Ports, relationships to Network Interface and Hosts</td>
<td>power_watts, btu_h, u_size, hw_eos, hw_eoes, host_created_date, host_modified_date, last_update_success Network Interface: interface_key, interface_netmask, interface_ipaddress, interface_subnet.</td>
<td>switch_ci, switch_host_rel, switch_interface_ci</td>
</tr>
<tr>
<td>si.xml</td>
<td>Software Instances: relationships to Hosts and BAIIs; Cyclic dependencies to: peer-to-peer communicating SIs, client-server communicating SIs, Contained SIs, DependedOn SIs</td>
<td>switch: switch_key, switch_name, description, status, model, ostype, osversion Switch Port: switch_port, interface_key, connected_ip_addr, connected_mac_addr, port_speed, port_duplex, port_negotiation, port_description, port_state, port_domain, port_vlan, port_vland_id Host: key Cyclic Dependency SIs: related SI key</td>
<td>si_ci, si_host_rel, si_client_server_comms_rel, si_contained_si_rel, si_dependend_on_si_rel, si_peer_to_peer_comms_rel, si_bai_rel</td>
</tr>
<tr>
<td>package.xml</td>
<td>Packages and Hosts with those packages installed</td>
<td>Software Instance: si_key, si_name, si_type, si_version, product_version, prod_release, edition, service_pack, build, patch, revision, si_eos, si_eoes BAI: bai_key Host: key</td>
<td>package_ci, package_host_rel</td>
</tr>
<tr>
<td>patch.xml</td>
<td>Patches and Hosts with those patches installed</td>
<td>Package: package_key, package_name, package_version, package_revision, package_os, package_created_date, package_modified_date Host: host_key</td>
<td>patch_ci, patch_host_rel</td>
</tr>
<tr>
<td>file.xml</td>
<td>Files and relationships to SIs and Hosts</td>
<td>Patch: patch_key, patch_name, patch_os, patch_created_date, patch_modified_date Host: host_key</td>
<td>file_ci, file_host_rel, file_si_rel</td>
</tr>
<tr>
<td>cluster.xml</td>
<td>Clusters and relationships to member Hosts</td>
<td>File: file_key, file_path, file_size, file_md5sum, file_last_modified, file_created_date Host: host_key SI: si_key</td>
<td>cluster_ci, cluster_host_rel</td>
</tr>
<tr>
<td>virtualcontainer.xml</td>
<td>Virtual Hosts and related Physical Hosts</td>
<td>Cluster: cluster_key, cluster_name, cluster_type, cluster_created, cluster_modified Host: host_key</td>
<td>virtualcontainer_ci, virtualcontainer_vhost_rel</td>
</tr>
<tr>
<td>hostcontainer.xml</td>
<td>Export Host Containers and their Host relationships</td>
<td>SI running virtualisation software: v_instance_key, v_instance_name, v_product_name, v_product_version, v_product_last_update_success, v_product_created, v_product_modified Running Host: v_host_key</td>
<td>hostcontainer_ci, hostcontainer_host_rel</td>
</tr>
</tbody>
</table>
Extended RDB mapping schema

The following diagram provides a description of the target schema.

This diagram illustrates an extended RDB mapping schema.

Configuration on the RDB side

To create this schema, the Extended RDB mapping set is provided with the following database specific scripts in the $TIDEWAY/java/integrations/mappings/extended-rdb directory:

- mySQL.sql
- Oracle.sql
- SQLServer.osql
- SQLServer.sql

There are also generic SQL statements in each of the mapping (xml) files for reference.

MySQL

In the MySQL command line client
use <databasename>
source <file>

Oracle

In a command window:
sqlplus <user>/<password>@<SID> -d <databasename>
@<filename>

SQL Server

In a command window:
osql -S <hostname>,<port> -U <user> -P <password> -i <file>
Export APIs

The BMC Atrium Discovery Export APIs enable users to interrogate the datastore using a script or program, and receive data back as a stream of text, an empty string, or a return code. The BMC Atrium Discovery Export APIs provide the following benefits for users:

- Ability to automate processes
- Ability to extract data for processing in third party applications

Previously users have been able to perform queries on the datastore through the Reports page and see the results displayed as HTML in the UI.

CSV API

The CSV API enables users to interrogate the datastore using a script or program, and receive data back as a stream of CSV (comma separated values) text, an empty string, or a return code.

Usage

The CSV API is intended to be used by a script or program. It can also be accessed using a browser, or using the wget command, though these would generally be used for testing. When using a URL you should ensure that the the entire URL is properly encoded. For example, special characters in the password might require encoding. The following section describes the parameters required and values returned when using the CSV API. The query is specified as a URL of the form:

```
http://appliance/ui/api/CsvApi?query=search_string&username=username&password=password
```

If you are using an appliance with https redirection enabled, you must use an https URL in your scripts. The redirection script changes the http request and the script will not work. To enable or disable https redirection, see Configuring HTTPS settings.

```
https://appliance/ui/api/CsvApi?query=search_string&username=username&password=password
```
The URL is entered on a single line. In these examples, appliance is the resolvable hostname or IP address of the appliance, search_string is a query string as recognized by the data store, username and password are the user name and password of the BMC Atrium Discovery user.

⚠️ Security Considerations

During a typical HTTP request to the CSV API, the username and password is visible in plain text and can be recovered in order to gain unauthorised access to the BMC Atrium Discovery user interface.
To mitigate this, follow two security best practices:

1. Always use a dedicated, read-only account for queries, and never use a high level account such as 'system'.
2. Always use https access so that the username and password are not in plain text.

💡 HTML URL Encoding

When constructing a URL string, care must be taken to properly encode characters that are not allowed in a URL, for example spaces:
https://appliance/ui/api/CsvApi?query=Search%20Host%20show%20name&username=username&password=password

A reference of encodings can be found here.

XML API

The XML API enables users to interrogate the datastore using a script or program, and receive data back as a stream of XML, an empty string, or a return code.

Usage

The XML API is intended to be used by a script or program. It can also be accessed using a browser, or using the wget command, though these would generally be used for testing. The following section describes the parameters required and values returned when using the XML API.

The query is specified as a URL of the form:

http://appliance/ui/api/XmlApi?query=search_string&username=username&password=password

If you are using an appliance with https redirection enabled, you must use an https URL in your scripts. The redirection script changes the http request and the script will not work. To enable or disable https redirection, see Configuring HTTPS settings.
The URL is entered on a single line. In these examples, appliance is the resolvable hostname or IP address of the appliance, search_string is a query string as recognized by the data store, username and password are the user name and password of the BMC Atrium Discovery user.

**Security Considerations**

With a normal HTTP request to the XML API the username and password will be visible in plain text and could be recovered in order to gain unauthorised access to the UI. To mitigate this there are 2 security best practises

1) Always use a dedicated read only account for queries, never use a high level account such as 'system'
2) Always use https access so that the username and password are not in plain text.

**HTML URL Encoding**

When constructing a URL string, care must be taken to properly encode characters that are not allowed in a URL, for example spaces:

```
https://appliance/ui/api/XmlApi?query=Search%20Host%20show%20name&username=username&password=password
```

A reference of encodings can be found here

**Parameters**

The following parameters are required for each script that interrogates the data store:

- A valid query string as recognized by the datastore.
  See the BMC Atrium Discovery Searching and Reporting Service Technical Note for details of the syntax of datastore queries.
- A valid user name and password.
  If you do not supply the correct credentials, the query fails.

**Note**

Unless you are using https, user names and passwords that are sent from a script are transmitted unencrypted.

- The CSV API accepts an optional Boolean parameter enabling you to specify whether column headings are to be returned with CSV data. This parameter is ignored by the XML API.
To specify that headings are returned, use `headings=1`, the default.
To specify that headings are not returned, use `headings=0`.

For example:

```
http://appliance/ui/api/CsvApi?query=SEARCH%20Host&username=system&password=system&headings=0
```

### Return values

The API returns the data and error codes:

- If the query is valid and successful, a block of CSV formatted text or XML is returned.
- If the query is valid but unsuccessful, that is, the datastore does not contain any matching data, an empty string is returned.
- If the query is empty, an error code of 1 is returned.
- If the CSV or XML formatter is not found, an error code of 2 is returned. If you receive this error, contact Customer Support.
- If there is a problem running the query an error code of 3 is returned. Check the syntax of the query by referring to the BMC Atrium Discovery Searching and Reporting Service Technical Note.
- If the login fails, an HTTP server response code of 401 (Unauthorised) is returned.

If the query takes longer to run than the appliance web server timeout, then no output is received by the script. By default, the timeout is ten minutes. If you need to increase the appliance web server timeout, contact Customer Support.

You might also experience client-side timeouts. See No Client Timeout for further information.

### Testing

This topic provides instructions for testing export APIs.

#### Browser-based testing

You can test the BMC Atrium Discovery Export APIs using a browser. Enter the following URL in a browser, replacing `appliance` with the resolvable host name or IP address, using the required `CsvApi` or `XmlApi` label, and by using the appropriate user name and password for the appliance that you are connecting to. For example, using the CSV API:

```
http://appliance/ui/api/
    CsvApi?query=SEARCH%20Host&username=system&password=system
```
A dialog is displayed. Click the **Open** button to view that data in the application registered for CSV files. Click the **Save** button to save the data. The default file name is **CsvApi**.

For example, using the XML API:

```plaintext
http://appliance/ui/api/
XmlApi?query=SEARCH%20Host&username=system&password=system
```

The results are displayed in the browser. Use the **File > Save** menus to save the output as an XML file.

**Command-line testing**

You can also test the BMC Atrium Discovery Export APIs from a UNIX command line using the `wget` command.

In this section the examples are for the CSV and XML APIs. The queries in all of the examples are appended to the following command line with no intervening spaces. For the CSV API:

```plaintext
$ wget http://appliance/ui/api/CsvApi?
```

For the XML API:

```plaintext
$ wget http://appliance/ui/api/XmlApi?
```

An example query to be appended to the command lines is shown as:

```plaintext
query=SEARCH%20Host&username=system&password=system
```

The `wget` command returns the HTTP status code, for a successful query this is **200 OK**. The data returned by the query above is written to a file in the current working directory which for the CSV API is called:

```plaintext
CsvApi?query=SEARCH%20Host&username=system&password=system
```

For the XML API, the file is called
Example Python script

The following Python script demonstrates similar tests to the ones described in the previous section. This is written for the CSV API but would be equally valid for the XML API.

```
import urllib

def runQuery():
    # Successful output
    query = "http://appliance/ui/api/CsvApi?query=SEARCH%20Host&username=system&password=system"
    f = urllib.urlopen(query)
    print f.read()
    # Empty query - output will be 1.
    query1 = "http://appliance/ui/api/CsvApi?username=system&password=system"
    f = urllib.urlopen(query1)
    print f.read()
    # Invalid query - output will be 3.
    query3 = "http://appliance/ui/api/CsvApi?query=SEARCHX%20Host&username=system&password=system"
    f = urllib.urlopen(query3)
    print f.read()
    # Authorisation failure - Output server code of 401.
    query4 = "http://appliance/ui/api/CsvApi?query=SEARCH%20Host&username=system&password=system1"
    f = urllib.urlopen(query4)
    print f.read()

if __name__ == '__main__':
    runQuery()
```

⚠️ With no exception handler around Query4, a 401 Unauthorised IOError traceback is produced.

Example queries

This topic provides examples of queries.

- Empty query
- No client timeout
- Invalid query
- Authorization failure

Empty query

The following example returns an error code of 1; this is the result of an empty query. The text to be appended to the command lines for this example is:
username=system\&password=system

In this case, the `query=SEARCH\%20Host` string is missing. Use the `cat` command to view the output file; it contains only the returned error code; a 1.

**No client timeout**

You might experience client-side timeouts; in this case you will encounter an unexpected end of file in the output. To avoid this, increase the client timeout to a value in excess of the web server timeout, or disable it entirely.

The following example shows how to disable the `wget` timeout. If the query takes longer to run than the appliance web server timeout, no output will be received.

```
$ wget --timeout=0 http://appliance/ui/api/...
```

**Invalid query**

The following example returns an error code of 3; this is the result of an invalid query. The text to be appended to the command lines for this example is:

```
query=SEARCH\%20Host\&username=system\&password=system
```

The query string now uses `SEARCHX` instead of `SEARCH`. This query will fail as there is no `SEARCHX` command defined in the query language. Use the `cat` command to view the output file; it contains only the returned error code; a 3.

**Authorization failure**

The following example shows an authorization failure; this is the result of an invalid password. The text to be appended to the command lines for this example is:

```
query=SEARCH\%20Host\&username=system\&password=system1
```

The query string now uses a password of `system1` instead of `system`. This causes an authorization failure returning a server response code of 401. This can be seen on the terminal the command was issued from.
Integrating

This section provides information on integrating other products with BMC Atrium Discovery. It contains the following sections:

- CMDB synchronization
- Enabling BMC Atrium Orchestrator to run commands
- BMC Atrium Orchestrator VMotion events

CMDB synchronization

CMDB synchronization provides a configurable mechanism to keep data in the BMC Atrium CMDB (CMDB) synchronized with information discovered by BMC Atrium Discovery.

In BMC Atrium Discovery 10.2, CMDB synchronization has been updated to provide the capability of connecting to multiple CMDB datastores. Connections to CMDBs are set up individually, enabling you to configure specific filters and blackout windows for each. You can also enable or disable and pause or resume synchronization on individual connections, or globally.

In previous releases the CMDB had to be queried to determine which updates BMC Atrium Discovery should make when syncing, and this step represented a large proportion of the overall time taken. BMC Atrium Discovery now maintains an authoritative model of the dataset of each configured CMDB connection, in its datastore. When a CMDB sync connection is created, BMC Atrium Discovery creates a shadow copy of the data that the corresponding CMDB dataset should hold and sends that to the CMDB. When data is modified as the result of discoveries, deletions or manual syncs, these changes are updated in the shadow copy and propagated to the CMDB, without the need to first check the contents of the actual CMDB.

Occasionally the model stored in the CMDB dataset becomes out of step with the shadow copy for a CMDB connection and requires resynchronization. For example, if CMDB tools have been used to modify the dataset. In this case, when updates are written to non-existent nodes, instance errors are raised. When BMC Atrium Discovery registers these instance errors, a resynchronization is recommended. In other cases such as after upgrade to version 10.2 or the creation of a new CMDB sync connection to a non-empty dataset, a resynchronization is mandatory, and CMDB sync is disabled until the resynchronization is complete.

Do not change the CMDB synchronization configuration at the same time as changing the cluster configuration.

- Changing the CMDB sync configuration means, adding connections, removing connections, and starting or stopping a resync.
- Changing the cluster configuration means adding members, removing members, moving the coordinator, or changing fault tolerance.
• Data model mapping
• Supported CMDB versions
• Deprecated versions of CMDB synchronization
• Synchronization stages
• Resynchronization
• Initial configuration
• To perform synchronization
• To prevent synchronization at certain times
• To filter synchronization data

### Starting from Technology Knowledge Update 2014-Jul-1, the CMDB sync mappings have been enhanced with two new capabilities:

#### Support for HasImpact population

Relationships in Atrium CMDB can be marked as being “impactful”, by setting the HasImpact attribute to Yes and the ImpactDirection attribute to a suitable value. A new CMDB sync data model means that BMC Atrium Discovery can now set these impact attributes directly, rather than relying on Atrium CMDB’s impact normalization rules. To enable the new data model:

1. In the Atrium Core Normalization application, use the Configuration Editor to disable the setting labelled Impact Normalization for the ADDM dataset
2. When adding or editing a connection in BMC Atrium Discovery, ensure the Data Model setting labelled "CMDB 7.6.03 and later (with impact attributes)" is selected (this is the default).

#### Simple static application models

BMC Atrium Discovery allows you define dynamic automatically-updating application models, using Collaborative Application Mapping and related features. That leads to robust comprehensive application models, but it does require time to investigate the application and define a suitable set of patterns. In some circumstances it is useful simply to manually group some hosts or software instances together to create a “static” application model. Such models are of course not automatically updated, but they are extremely quick and easy to define.

To define a static application model, add Hosts, SoftwareInstances or LoadBalancerServices to a group in BMC Atrium Discovery, and give the group a name that starts with “StaticApp:”, in exactly that form. Do not use sub-groups. That causes the creation of a BMC_Application CI in Atrium CMDB, named after the group. For example, a group named “StaticApp: Example Application” results in a BMC_Application CI with
name “Example Application”. As with all groups, the StaticApp group may contain any node kind, but only Host, SoftwareInstance and LoadBalancerService nodes will actually be related to the BMC_Application CI in the CMDB.

Data model mapping

The BMC Atrium Discovery data model (Discovery model) is different from the Common Data Model (CDM) used in the CMDB, so the synchronization mechanism is responsible for transforming the required information from one data model to the other. The Discovery model is known as the source model, and the CDM is referred to as the target model.

Both models are graph models, meaning that they represent data as nodes connected to each other with relationships. Synchronization operates on portions of the full graph known as subgraphs. A subgraph contains a root node (such as a host computer), plus all the related nodes that belong to it (such as interface information, software information, and so on), referred to as its components. Some nodes can be shared, meaning that they belong to more than one subgraph. For example, the node representing a subnet is shared by all the host computers on that subnet.

The mapping between the Discovery model and the CDM is defined in syncmapping definitions within TPL files.

The Default CDM Mapping provides a comprehensive mapping from the Discovery model to a best-practices CDM model. If you have added custom nodes to Discovery they are not exported automatically by the default CDM mapping, instead, you will have to create additional mapping definitions within TPL files.

Supported CMDB versions

At the time of its release, integration of the current version of BMC Atrium Discovery was supported with the following versions of BMC Atrium CMDB:

- 9.0 – Fully supported. The integration is fully supported, though formal BMC certification is pending. There are some performance issues between BMC Atrium Discovery 10.2 and CMDB 9.0. These are resolved in CMDB 9.0 SP1.
- 8.1.02
- 8.1.01
- 8.1.00
- 8.0.00
- 7.6.04

If you intend to use CMDB sync, BMC recommends that you use BMC Atrium Discovery version 10.2.0.2 or later.

For more information, check the BMC Solution and Product Availability and Compatibility utility (SPAC) (support login required).
 Deprecated versions of CMDB synchronization

BMC Atrium Discovery version 8.1 introduced the first automatic integration using CMDB synchronization. At that time the previously used method (the Atrium CMDB adapter) was deprecated. In BMC Atrium Discovery 9.0, the Atrium CMDB adapter was removed.

Synchronization stages

Synchronization occurs in the following stages:

1. A root source node is chosen for synchronization. The root node kinds are Host, MFPart, NetworkDevice, Printer, SNMPManagedDevice, and Storage System corresponding to the main kinds of device that can be discovered. The list of possible root nodes is fixed and cannot be altered in the Syncmappings.
2. The source subgraph is populated by retrieving all the relevant data from the Discovery datastore.
   - The ADDM filter sub page for a CMDB connection can be used to specify which source nodes will be synced.
3. The source subgraph is transformed into a target subgraph, centered around a target root node.
   - The CMDB filter sub page for a CMDB connection can be used to specify which CIs will be synced, that correspond to the nodes allowed through by the ADDM filter.
4. The target subgraph is compared with the corresponding subgraph stored in the Discovery datastore for the CMDB connection.
5. The updates identified are made in batches to the CMDB. Once an update has been made to the CMDB (by creating, updating and (soft) deleting CIs and relationships), it is also made to the local shadow copy for the connection. The next batch is then processed in the same way until no more updates are required.

⚠️ Note

If any changes are made to the CDM (for example, adding attributes), you cannot synchronize to those attributes until the Tideway service has been restarted. After the tideway service starts, the CDM is read and all customized classes and attributes are available for CMDB synchronization.

Resynchronization

Occasionally the model stored in the CMDB dataset becomes out of step with the shadow copy for a CMDB connection and require resynchronization. For example, if CMDB tools have been used to modify the dataset. In this case, when updates are written to non-existent nodes, instance errors
are raised. When BMC Atrium Discovery registers these instance errors, a resynchronization is recommended. In other cases such as after upgrade to version 10.2 or the creation of a new CMDB sync connection to a non-empty dataset, a resynchronization is mandatory, and CMDB sync is disabled until the resynchronization is complete.

Resynchronization must assess the contents of the CMDB dataset against the authoritative version stored in the Discovery shadow copy for the CMDB connection and send the appropriate updates to the CMDB. The stages in a resynchronization are:

1. Preparation – Prepare Resync performs the following actions, without changing any data in the CMDB:
   a. Reads identities of all CIs and relationships from the CMDB dataset.
   b. Transforms all BMC Atrium Discovery data into the local shadow copy.
   c. Compares the data read from the CMDB with the updated local shadow copy.

2. Commit – Commit Resync performs the following actions in the CMDB:
   a. Creates CIs and relationships missing from the dataset.
   b. Marks deleted any CIs or relationships present in the dataset but not in the local shadow copy.
   c. Updates all CIs and relationships to ensure their attributes match those in the local shadow copy.

In the following situations a resynchronization is mandatory, CMDB sync is disabled until the resynchronization is complete.

**Upgrade**

When you upgrade an appliance which is configured for CMDB sync, its configuration is modified to the new format, though the shadow copy is not populated. However, the CMDB dataset corresponding to the connection may still be populated from a sync performed before the upgrade. If a sync were attempted immediately, BMC Atrium Discovery would not be aware of the existing contents of the dataset, and the result could be the creation of duplicate CIs. A resynchronization is therefore required, to query the entire contents of the dataset before updating it. Before the resynchronization can be performed though, it may be necessary to manually create filters corresponding to the ones that existed prior to the upgrade - see CMDB Synchronization Filters.

When you add a new sync connection, its shadow copy is created, but is empty. When the connection is created, the credentials and CMDB version are checked. The first check also determines whether the CMDB dataset is populated. If so, a resynchronization is required to populate the shadow copy with discovered information and sync that with the CMDB. If the dataset and shadow copy are both empty then no resynchronization is required.

**Failed or cancelled resynchronization**

If a resynchronization fails or was cancelled, a resynchronization is required before the connection can be resumed.
Initial configuration

The following steps are required to set up CMDB synchronization:

- Preparing BMC Atrium CMDB for synchronization
- Setting up a CMDB synchronization connection

To perform synchronization

Synchronization can be configured to run in a *continuous* mode, meaning that as soon as BMC Atrium Discovery completes its scan of a device, the data corresponding to it is queued for synchronization with the CMDB. Alternatively, synchronization can be launched manually for one or more devices from within the browsing interface.

- Managing CMDB synchronization
- Batch and individual synchronization

To prevent synchronization at certain times

Typically, synchronization can occur at any time. However, synchronization can place a substantial load on the CMDB, so it might be necessary to prevent synchronization from occurring at times the CMDB will be used heavily. This can be achieved by configuring Configuring CMDB Sync blackout windows.

To filter synchronization data

By default, all details about all devices are synchronized to the CMDB. In some situations, it might be useful to only synchronize a subset of the data. After each of the first three synchronization stages, the data can be filtered to affect the data that finally reaches the CMDB. See Filtering nodes and CI types for CMDB Sync for more information.

Preparing BMC Atrium CMDB for synchronization

At the time of its release, integration of the current version of BMC Atrium Discovery was supported with the following versions of BMC Atrium CMDB:

- 9.0 – Fully supported. The integration is fully supported, though formal BMC certification is pending. There are some performance issues between BMC Atrium Discovery 10.2 and CMDB 9.0. These are resolved in CMDB 9.0 SP1.
- 8.1.02
- 8.1.01
- 8.1.00
- 8.0.00
- 7.6.04

If you intend to use CMDB sync, BMC recommends that you use BMC Atrium Discovery version 10.2.0.2 or later.
For more information, check the **BMC Solution and Product Availability and Compatibility utility (SPAC)** (support login required).

**Before you begin**

Before you can synchronize data to a supported BMC Atrium CMDB version, you must complete the following required tasks in the given order:

---

**Integrating with earlier versions of BMC Atrium CMDB**

BMC Atrium CMDB versions earlier than 7.6.04 are no longer supported and are not recommended for use. Synchronizing data to these earlier versions required you to complete some additional tasks. For more information, see the **BMC Atrium Discovery 8.3** documentation.

1. **Create the BMC.ADDM dataset**: The BMC.ADDM dataset must be manually created in the CMDB before a synchronization is attempted.

2. **Create the Job to merge the BMC.ADDM dataset with BMC.ASSET**: After the BMC.ADDM dataset has been created, you must then create the job to reconcile it with the BMC.ASSET dataset.

3. **Check the BMC.ADDM dataset configuration**: When using ITSM 7.0 with a backwards compatibility patch, you also need to ensure that the BMC.ADDM dataset is trusted.

---

After completing the preceding tasks, you can start synchronizing BMC Atrium Discovery data to BMC Atrium CMDB.

**Performance considerations**

To obtain the maximum synchronization performance when using CMDB synchronization with BMC Atrium Discovery version 8.3.00 and later, you should consider tuning the database which BMC Atrium CMDB (or BMC Remedy AR System) is using. For more information, see the following documentation corresponding to your product version:

- **BMC Remedy AR System 8.1 Performance tuning for Business Service Management**
- **BMC Remedy AR System 8.0 Performance tuning for Business Service Management**
- **BMC Remedy AR System Server 7.6 Performance Tuning for Business Service Management White Paper**

The following sections in the White Paper are particularly relevant:

- Tuning an Oracle server.
- Best practices for tuning Oracle database servers.
- Tuning a SQL Server database.
- Best practices for tuning SQL Server database servers.
To create the BMC.ADDM dataset

Creating a **BMC.ADDM** dataset provides a dataset into which BMC Atrium Discovery is exported before being taken into the BMC Atrium CMDB. You create the dataset by using the BMC Atrium CMDB Reconciliation console. It is recommended that you create the BMC.ADDM dataset on the primary AR System server.

**Warning**

The dataset you are creating is intended to receive data from *only* one BMC Atrium Discovery appliance. If your deployment architecture requires that several scanners or consolidator synchronize to the same CMDB, then you must create a separate dataset for each. For example, **BMC.ADDM1**, **BMC.ADDM2**, and associated reconciliation jobs.

Once created, do not make any changes to any CMDB synchronization dataset.

1. You can create a dataset from one of two places in the console.
   - From the Identify activity, click **Add Dataset Identification Group Association**, and then click **Create Dataset**.
   - From the Mid Tier, Open Atrium Core Console > Applications > Reconciliation console, and then click **Create Dataset**.
2. Complete the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name for the dataset. Usually set to <strong>BMC.ADDM</strong>.</td>
</tr>
<tr>
<td>ID</td>
<td>The system identifier for the dataset. This must match the ID used in the configuration within BMC Atrium Discovery. The default is <strong>BMC.ADDM</strong>.</td>
</tr>
<tr>
<td>Accessibility</td>
<td>Set to &quot;Writable&quot;.</td>
</tr>
<tr>
<td>DatasetType</td>
<td>Set to &quot;Regular&quot;.</td>
</tr>
</tbody>
</table>

3. Click **Save**.

For more information, see:

- Defining a Precedence Association Set for reconciliation Merge activities for BMC Atrium Core 8.1
- Defining a Precedence Association Set for reconciliation Merge activities for BMC Atrium Core 8.0
- BMC Atrium CMDB 7.6.04 Normalization and Reconciliation Guide
To create the Identification, Merge and Purge Job

BMC Atrium Discovery exports its data to a staging dataset in BMC Atrium CMDB. The content of this dataset should then be reconciled into the BMC.ASSET dataset. To accomplish this, a new Job must be added in the BMC Atrium CMDB Reconciliation Console. It is recommended that you create the Identification, Merge and Purge Job on the primary AR System server.

To add the Identification, Merge and Purge Job

1. From the Mid Tier, Open BMC Atrium Core Console > Applications > Reconciliation Application.
2. Select the Create Standard Identification and Merge Job Icon
3. Give the job a meaningful name (for example, "BMC ADDM - Identification, Merge and Purge").
4. Set the Source Dataset to be "BMC.ADDM".
5. After the job is Created, select BMC ADDM - Identification, Merge and Purge Job.
6. Click the Edit Job icon.
7. From the Job Editor window, in the Activities pane, select the New tab.
8. In the New Activity pane, select Type as 'Purge', Sequence as '700' and Name as BMC ADDM - Purge.
9. In the Datasets option, select Datasets as BMC.ADDM.
10. Click Done.
11. Click Save in the Job Editor.

⚠️ Note

The recommended practice is to create a precedence group that specifies a precedence for the BMC.ADDM dataset. The value depends on your relative preference for data populated by BMC Atrium Discovery and other data providers to the CMDB. For more information about how precedence groups are configured, see the following documentation corresponding to your product version:

- Defining a Precedence Association Set for reconciliation Merge activities for BMC Atrium Core 8.1
- Defining a Precedence Association Set for reconciliation Merge activities for BMC Atrium Core 8.0
- BMC Atrium CMDB 7.6.04 Normalization and Reconciliation Guide – section named "Defining a Precedence Association Set".
To check the BMC.ADDM dataset configuration

When using BMC Remedy ITSM 7.0 with a backward compatibility patch, you also need to ensure that the **BMC.ADDM** dataset is trusted. To do this:

1. Access the Home page from BMC Remedy User or a browser.
2. Access the Application Administration Console.
3. Go to the Custom Configuration page.
4. From Foundation, select **Products / Operational Catalogs**.
5. Select **Trusted Datasets** for Product Catalog and click **Open**.
6. Select the **BMC.ADDM** dataset for Dataset Name.
7. Click **Save**.

To prepare synchronization to another CMDB

There might be occasions where you have already synchronized data to a CMDB, and you want to push data to a different CMDB. For example, if you are migrating from a development to a production environment for your CMDB, you might want to populate your new CMDB with fresh inventory data, independent of previous experiments you might have performed in the development instance. To synchronize the data to the new CMDB, perform the following steps using continuous CMDB synchronization.

1. Stop any discovery runs.
2. Stop continuous synchronization (wait for it to finish processing the queue).
3. Change your configuration to the new CMDB dataset (see the tasks to complete before you begin).
4. From the **CMDB Sync > Configuration** tab, click **Test Connection** to the new CMDB dataset.
   Ensure that the connection is successful before proceeding.
5. From the **Infrastructure > Hosts** page, send one or two hosts to the CMDB, and select **Actions > Sync with CMDB**.
6. Start continuous synchronization.
7. Restart your discovery runs.

The data is inserted into the new CMDB dataset as your Discovery jobs are executed. BMC Atrium Discovery detects that the data does not exist, and then inserts the data.

To synchronize data from multiple BMC providers to the CMDB

When multiple BMC providers synchronize data to the CMDB (such as BMC Atrium Discovery or BMC BladeLogic), there is the possibility that BMC_OperatingSystem instances might not be merged during reconciliation.
BMC Atrium Discovery is an agentless discovery tool and, as such, it might retrieve slightly different information than an agent-based product (for example, "SunOS" or "Solaris" for the same server). To eliminate this problem, you can add a fallback reconciliation rule based on the ClassId for BMC_OperatingSystem instances.

**Synchronizing functional components**

Collaborative Application Mapping (CAM) represents an application with three levels: `SoftwareInstances` are combined into `Functional Components`, which are combined into a `Business Application Instance`. In previous releases with TKU older than `TKU 2014-Jan-1`, CAM models were synchronized to the CMDB, the models were represented with two levels, with `BMC_SoftwareServer` CIs grouped into a `BMC_Application CI`.

BMC Atrium Discovery version 10 now synchronizes Functional Components to `BMC_ConcreteCollection` CIs. For details of the mappings, see, [CDM Mapping for Host](#) and [CDM Mapping for MFPart](#).

**Setting up a CMDB synchronization connection**

This topic provides instructions for configuring synchronization from BMC Atrium Discovery to one or more instances of BMC Atrium CMDB.

- **Before you begin**
- **CMDB synchronization and clusters**
- **To configure a CMDB synchronization connection**

**Before you begin**

Before you can synchronize from BMC Atrium Discovery to any instance of BMC Atrium CMDB, you must have prepared that CMDB.

**CMDB synchronization and clusters**

When configuring CMDB synchronization from a cluster, you can use any machine for the initial configuration and subsequent updates. All data synchronized to any connected BMC Atrium CMDB is sent from the coordinator.

Do not change the CMDB synchronization configuration at the same time as changing the cluster configuration.

- Changing the CMDB sync configuration means, adding connections, removing connections, and starting or stopping a resync.
- Changing the cluster configuration means adding members, removing members, moving the coordinator, or changing fault tolerance.
To configure a CMDB synchronization connection

1. From the Model section of the Administration page, click CMDB Sync.
   - If no CMDB Sync has been configured, a banner displays stating that the no CMDB Sync connections are configured.

![CMDB Sync banner](image)

Click Add CMDB Connection to configure the first connection to a BMC Atrium CMDB server.
- If CMDB Sync connections exist, click Add CMDB Connection to configure another connection to a BMC Atrium CMDB server.

![Add CMDB Connection](image)

a. i. Enter the following information on the Add CMDB Connection page:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the CMDB Sync connection.</td>
</tr>
<tr>
<td>Network Address</td>
<td>The BMC Atrium CMDB host. This can be specified as one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Hostname or FQDN</td>
</tr>
<tr>
<td></td>
<td>• IPv4 or IPv6 address</td>
</tr>
</tbody>
</table>

- If BMC Atrium CMDB is installed with an AR System server group, you must enter the following based on the high availability status of the server group:
  - (If high availability is configured) The host name or IP address of the load balancer.
  - (If high availability is not configured) The host name or IP address of the primary AR System Server.

To learn about the AR System server groups configuration, see the corresponding documentation for versions 8.1, 8.0, and 7.6.04.

To learn about the AR System server high availability configuration and hardware load balancer, see the corresponding documentation for versions 8.1, 8.0, and 7.6.04.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify TCP Port</td>
<td>The communication port. BMC Atrium CMDB typically uses a portmapper service to automatically choose a suitable communication port. If this is not appropriate in your environment, you can configure the CMDB to listen on a specific port, and then specify that port in this field.</td>
</tr>
<tr>
<td>Username</td>
<td>The username of the BMC Atrium CMDB user that is at least a member of a group having the CMDB Data Change All role. If in doubt, create a dedicated discovery user with the same profile as the standard CMDB Demo user.</td>
</tr>
</tbody>
</table>

- The CMDB Demo user permissions at least are required for multitenancy to work. A user is not permitted to connect to BMC Atrium CMDB from a second IP address within 4 hours of their last activity at the first IP address. For a failover scenario, you could also create a second discovery user to connect from the failover appliance.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>The password of the CMDB user (or blank if the user has no password).</td>
</tr>
<tr>
<td>Continuous Sync</td>
<td>Select the checkbox to enable continuous sync for this CMDB connection. During continuous CMDB synchronization, whenever BMC Atrium Discovery finishes scanning a device, it is added to the synchronization queue. Similarly, whenever a device node is removed due to aging, it is also queued for synchronization, meaning that the deletion is propagated into the CMDB. In this way, the CMDB stays completely synchronized with BMC Atrium Discovery.</td>
</tr>
<tr>
<td>Batch Size</td>
<td>Available from BMC Atrium Discovery version 10.2.0.2. Enables you to configure the batch size used for batch operations. If you experience timeouts during a batched operation, most likely during a resync, these may be prevented by setting it to a lower value. You can set the batch size to anything between 1 and 1000. The default is 100, which was the fixed batch size in previous releases.</td>
</tr>
<tr>
<td>Resync Threads</td>
<td>Available from BMC Atrium Discovery version 10.2.0.2. Enables you to configure the number of threads used by the resync process to perform creates, updates, and deletes in the CMDB. Higher values can give quicker resyncs, though this does add more load to the CMDB system. You can set the number of resync threads from 1 to 20. The default is 1.</td>
</tr>
<tr>
<td>Dataset ID</td>
<td>The ID of the dataset used for Discovery data. The default is BMC.ADDM.</td>
</tr>
<tr>
<td>Data Model</td>
<td>The data model for your CMDB. Different versions of the CMDB have different data models, so the data from BMC Atrium Discovery must be transformed differently according to the CMDB version. The preferred value for Data Model is <strong>CMDB 7.6.03 and later (with impact attributes)</strong>. Choose the correct data model from the menu. If you choose an incorrect value, you might encounter errors during synchronization.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data model value</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDB 7.6.03 and later</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate</td>
</tr>
<tr>
<td>(with impact attributes)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMDB 7.6.03 and later</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy BMC Service Impact Management version 7.4.</td>
</tr>
<tr>
<td>(with impact relationships)</td>
<td></td>
<td>BMC_Impact relationships with Name “ImpactOnly” are created. See warning below</td>
</tr>
<tr>
<td>CMDB 7.6.03 and later</td>
<td>7.6.03 and later</td>
<td>No impact details are set by ADDM. They may be set by Impact Normalization in the CMDB</td>
</tr>
<tr>
<td>(no impact details)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMDB 7.6 (deprecated)</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created</td>
</tr>
<tr>
<td>CMDB 7.5 (deprecated)</td>
<td>7.5</td>
<td></td>
</tr>
</tbody>
</table>

- **Do not use mixed modes with CMDB 7.6.03 and later (with impact relationships)**
- **If you specify the CMDB 7.6.03 and later (with impact**
1. **Field Description**

    Option, CMDB synchronization creates impact relationships as a relation (IaaR). Normalization in the CMDB using the default setting of `auto impact` creates Impact relationships as an attribute (IaaA).

    You must change your CMDB Normalization setting so that it does not use `auto impact` when the Data model is set to CMDB 7.6.03 and later (with impact relationships) in BMC Atrium Discovery. Do not use a mixed mode of (IaaA) and (IaaR).

**Multitenancy**

Multitenancy support. Selecting this check box enables you to choose a default company name from the dropdown list to assign to a discovery run during an initial scan. The list of companies is not populated until one connection has been successfully tested, or you click Get Companies (which also tests the connection).

- **i** This section is always displayed. However, if another connection exists, instead of the multitenancy configuration options, a message stating `Configure multitenancy from the overview page` is displayed along with a link to the CMDB Sync page.

**i** In BMC Atrium Discovery 10.2.00, multitenancy was configured in this screen whether or not connections existed. In BMC Atrium Discovery 10.2.0.1 and later, multitenancy is configured on the CMDB Sync home page.

2. Click Test Connection to verify that the Atrium CMDB server can be reached, and that the credentials are valid. Testing the connection also retrieves the list of companies from the CMDB.

- **i** Ensure that the connection is successful before proceeding.
Configuring Multitenancy

*Multitenancy* refers to a situation where a single application on a server supports multiple client organizations. In BMC Atrium Discovery, this means the ability to scan the estates of multiple organizations, and to assign a **company name** attribute to that scan. Any CMDB CIs created as a result of that scan can be assigned the appropriate company name.

Multitenancy is a global setting, it is either enabled or disabled. When you configure multitenancy you select a default company. That becomes the default for all new discovery runs, though you can choose to set any other company name, or none (no company).

The company name is assigned to the **cdm_company** attribute on the **Discovery Run node**. This is mapped onto CMDB CIs created as a result of that scan and assigned to the **Company** attribute.

**Important considerations when using multitenancy**

Multitenancy obtains company names from BMC Remedy ITSM through BMC Atrium CMDB. Therefore, to obtain company names, you must have a working **CMDB Sync** connection and BMC Remedy ITSM must be installed. You do not need to have performed a synchronization, but the connection must be configured and have been successfully tested, with a user with at least CMDB Demo user permissions. Once this is set up, the **Lookup Companies** dropdown is populated.

The company list on the appliance is not automatically refreshed from the BMC Atrium CMDB. If you do not see a company that you are expecting to see, refresh the list by clicking **Test Connection** on the CMDB Sync connection page. Similarly, if you add a new company name to BMC Remedy ITSM, you must refresh the list by clicking **Test Connection** on any CMDB Sync connection page.

If there is no company information available from ITSM, the Default Company does not contain any company names. Instead, it displays No Company.

**Multitenancy in Consolidation deployments**

In consolidating systems with multiple Scanning appliances feeding a Consolidation appliance, it is the Consolidation appliance that requires the CMDB sync connection. Scanning appliances should not be configured to synchronize with the CMDB, they should send data to the Consolidation appliance. As part of the communication that occurs between associated Consolidation and Scanning appliances, the company list on the Consolidation appliance is automatically populated to the Scanning appliances. However, the list on the consolidating appliance is not automatically refreshed from the BMC Atrium CMDB.

When you are working on a Scanning appliance, and you do not see a company that you are expecting to, you must log onto the Consolidation appliance and refresh the list on the Consolidation appliance by clicking **Test Connection** on any CMDB Sync connection page.
To enable multitenancy

1. From the Model section of the Administration page, click CMDB Sync.
2. The CMDB Sync page is displayed. If CMDB Sync connections exist, they are shown here.

To disable multitenancy

1. From the CMDB Sync page, click the Multitenancy Enabled: Company Name link.
2. Deselect Enabled and click Apply.
3. The CMDB Sync page is displayed, and the Multitenancy Enabled: Company Name link is replaced with Multitenancy Disabled.

Managing a CMDB sync connection

Additional configuration and management tasks can be performed on individual CMDB sync connections. These are accomplished from the Manage CMDB Connection page:
To view the Manage CMDB Connection page, click the name of a CMDB sync connection on the CMDB sync home page.

An Actions drop down enables you to perform the following actions on the connection:

- Delete the CMDB Connection
- Enable or Disable Continuous Sync
- Pause or Resume Sync
- Disable or Enable Sync
- Reset counters

The Manage CMDB Connection page is displayed showing the status tab. The other tabs are introduced below:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Details</td>
<td>Provides an editable form where you can edit the details used when the connection was created. See setting up the CMDB synchronization connection for information on the details you can edit.</td>
</tr>
</tbody>
</table>
| Filtering | Provides a two stage filter which enables you to first filter the root nodes, using the BMC Atrium Discovery model, to sync to the CMDB and then perform a second filter on the CIs in the generated model. This page has two sub pages accessed using the following buttons:  
  - ADDM – Enables you to select the BMC Atrium Discovery root device nodes used to create the model to synchronize to the CMDB. All device nodes are synchronized unless a filter is applied. You can create a filter for each root node kind, which may be:  
    - All  
    - None  
    - Matching a condition created using a three column selector  
  - CMDB – All device nodes matching the ADDM filter are synchronized to the CMDB. The CMDB filter enables you to add a second level of filtering on the data once it has been transformed into the Common Data Model. You can create a filter on each CI type, which may be:  
    - All  
    - None  
    - Matching a condition created using a three column selector |
You can check which root nodes will be allowed through the filter specified on the ADDM filter page, by following the link at the bottom of that page labelled "Click here to see the current candidate set". After selecting node(s) from the list that appears, and clicking the CMDB Sync Preview button above the list, a Software Context view of CIs corresponding to those nodes can be displayed, together with highlighting of any CIs filtered by the settings on the CMDB filter page. The highlighting distinguishes explicitly filtered CIs from those that are implicitly filtered (only connected to the model by explicitly filtered nodes).

CMDB synchronization can occur at any time by default. To prevent CMDB access for a connection during sensitive times, you can configure blackout windows. During a blackout window, all processing of the CMDB synchronization queue for that connection is paused. New nodes can still be scheduled for synchronization, by continuous or manual synchronization, but no processing occurs until the blackout window ends.

For more information on blackout windows, see Configuring CMDB Sync blackout windows.

Occasionally the model stored in the CMDB dataset becomes out of step with the shadow copy for a CMDB connection and require resynchronization. For example, if CMDB tools have been used to modify the dataset. In this case, when updates are written to non-existent nodes, instance errors are raised. When BMC Atrium Discovery registers these instance errors, a resynchronization is recommended. In other cases such as after upgrade to version 10.2 or the creation of a new CMDB sync connection, a resynchronization is mandatory, and CMDB sync is disabled until the resynchronization is complete.

For more information on resyncing a connection, see Resyncing a CMDB connection.

Filtering nodes and CI types for CMDB Sync

Nodes which are created, deleted, or updated are added to the CMDB sync queue. As not all of the discovered data is required in the CMDB, it is filtered to reduce the amount of data synced, and to increase the relevance of that data for CMDB users. Filtering is a two stage process, firstly the root nodes are selected and a filter applied, then, after the data is transformed into the Common Data Model, an additional CMDB CI filter is used to select those CIs to be synced to the CMDB.

- Filtering in the CMDB Synchronization workflow
- Filtering the root nodes
- Filtering CMDB CI types
- Filter Conditions
Filtering in the CMDB Synchronization workflow

Filters defined for CMDB sync are used during several stages of the CMDB synchronisation process:

The ADDM filter is applied to the *source subgraph* that is populated by retrieving all the relevant data from the Discovery datastore. This trims the data to avoid populating CMDB with excessive or unnecessary information.

After that source subgraph is transformed into the Common Data Model (a *target subgraph*, centered around a *target root node*), and the second filter is applied to the resulting data.

Filtering the root nodes

On the Filtering > ADDM pane of the CMDB Sync page, you can create a filter for each BMC Atrium Discovery root device node kind (which can select all, none, or nodes matching specific filter conditions) and preview the results when using a candidate node set (ensure that the candidate node set is visible; use Click here to see the current candidate set if necessary).

The defined filter creates the model of the candidate set that is synchronized to the CMDB later on. All device nodes are synchronized unless a filter is applied.

In the filter settings you can specify one of the following options:

- All nodes of the selected kind are synced to the CMDB
- None nodes of the selected kind are synced to the CMDB
- Only nodes of the selected kind which match the defined conditions are synced to the CMDB

Remove an individual definition by clicking its link then using the Clear selected ADDM filter definition button, or reset all filters using Clear All ADDM filter definitions.

Filtering CMDB CI types

All device nodes matching the ADDM filter created in Filtering the root node are synchronized to the CMDB, however the CMDB filter enables you to add a second level of filtering on the data once it has been transformed into the Common Data Model.

On the Filtering > CMDB pane of the CMDB Sync page, you can create a filter for each CI class (which can select all, none, or nodes matching specific filter conditions - see below), and preview the results on the ADDM filter page. All BMC Atrium CMDB CI classes are synchronized unless a filter is applied.

In the filter settings for the CI class you can specify one of the following options:

- All CIs of the specified class are synced to the CMDB
• No CIs of the specified class are synced to the CMDB
• Only CIs of the specified class which match the defined conditions are synced to the CMDB

Remove an individual definition by clicking its link then using the Clear selected CMDB filter definition button, or reset all filter definitions using Clear All CMDB filter definitions.

Filter Conditions
You can define conditions in the filtering panes reached from the ADDM or CMDB buttons, for BMC Atrium Discovery root nodes, or CMDB CI classes respectively. A filter can be constructed from these conditions, which pushes just the matching root nodes or CIs to the CMDB. Conditions can be defined for simple attribute values, or attributes of a destination node or CI reached by following a relationship, or logical combinations of either. The filters are shown in map form in the filter builder.

• Query Constructor
  • Example of creating a root node filter

Query Constructor
Query constructor contains two areas, where you specify the attributes that will be used in a query, and define the filtering conditions and values to match.

On the first area you can select an attribute to filter on, or follow relationships to other nodes from which you can select attributes to filter on, and define the conditions and the value upon.

Relationships are displayed with a following right arrow (>). To select the necessary attribute, you can scroll through the list or use the lookup tool beneath each pane.

When you enter text into the lookup tool, a drop down list of matches is displayed, from which you can select the attribute or relationship to use.

Clicking an attribute, for example OS Class, adds a Host: OS Class entry to the filter.

Clicking a relationship, for example Software Instance: Software Instances running on this host, populates the next pane to the right with the attributes and relationships of the related node. When you traverse to more than three relationships, it adds new pane and scrolls the previous panes to the left, hiding the first pane(s). You can scroll back by clicking the arrow to the left of the filter builder.

The second area provides a map of the filter that you are constructing.
You can evaluate conditions on an attribute or group of attributes using the following options:

- **All** — True when ALL conditions are true.
- **Any** — True when any of the conditions are true.
- **None** — True when none of the conditions are true.

For example, on the following screen, the Host node should match both conditions to be synchronised to the CMDB:

![Example of creating a root node filter](image)

**Example of creating a root node filter**

The following example shows how to use the filter builder to create a filter which will sync only those hosts in the organization that are web servers. In the case, you might want to find all Windows hosts running IIS and all UNIX hosts running Apache.

1. Create a filter for all Windows hosts running IIS:
   a. Select **Host** as an ADDM node kind to filter.
   b. Click the **Only Host nodes which match the following conditions**... radio button.
   c. Select **OS Class** from the list of the node attributes, and define a condition where **OS Class contains word "Windows"**.
   d. Click the **Add Condition** icon next to the OS Class filtering condition to arrange a new conditions evaluation container (with All, Any, or None options).
   e. Switch the focus inside the container you just created by clicking on that container's area.
   When the container is selected, it is shown highlighted in blue.
   f. To add the Software Instance filter, select **Software Instance: Software Instances running on this host**, select the **Name** attribute, and define a condition where **Name contains word "IIS"**.
   g. Click **Save condition**.

2. Create a filter for all UNIX hosts running Apache:
a. Switch the focus back to the root container by clicking it.
b. Select OS Class from the list of the node attributes, and define a condition where OS Class contains word "UNIX".
c. Click the Add Condition icon next to the OS Class filter you just created.
d. Switch the focus inside the container you just created by clicking on that container's area.
e. To add the Software Instance filter, select Software Instance: Software Instances running on this host, select the Name attribute, and define a condition where Name contains word "Apache".

3. Change the root condition to say Host where Any of these conditions are true (default is All).

4. Click Save condition.

The resulting filter is provided below:

![Filter result image]

Previewing the filtered results

List the filtered root nodes using the Click here to see the current candidate set button in the Filtering > ADDM pane of the CMDB Sync page.

Then use the CMDB Sync Preview button (located above the candidate set) to see a Software Context view of the CMDB model of selected nodes from the candidate set. The preview includes the explicitly filtered nodes and the implicitly filtered nodes, those only connected to the model by explicitly filtered nodes.

Importing and Exporting filters

You can use the same filter with multiple CMDB connections by exporting and importing filtering definitions. These actions are available on the top of the Filtering tab on the CMDB Sync page.
Configuring CMDB Sync blackout windows

CMDB synchronization can usually occur at any time. To prevent CMDB access for a connection during sensitive times, you can configure blackout windows to occur. During a blackout window, all processing of the CMDB synchronization queue for that connection is paused. New nodes can still be scheduled for synchronization, both by continuous synchronization and batch synchronization, but no processing occurs until the blackout window ends.

To configure a CMDB Sync blackout window

1. Click the name of a CMDB sync connection on the CMDB sync home page.
   The Manage CMDB Connection page is displayed
2. On the CMDB Sync connection page, click the Blackout Windows tab.
3. To add a new blackout window, click Add.
   The Add a New Blackout Window dialog is displayed.
4. Enter the information for the blackout window in the fields described in the following table.

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment</td>
<td>Enter a comment for the blackout window. Where the blackout window is referred to in the UI, this label is shown.</td>
</tr>
<tr>
<td>Frequency</td>
<td>Select a frequency for the blackout window. For example, this can be:</td>
</tr>
<tr>
<td></td>
<td>• Weekly by days of week</td>
</tr>
<tr>
<td></td>
<td>• Once per week</td>
</tr>
<tr>
<td></td>
<td>• Monthly by day of month</td>
</tr>
<tr>
<td></td>
<td>• Monthly by week of month</td>
</tr>
<tr>
<td>Start</td>
<td>Based on the selected frequency, the following options for starting the blackout window is displayed:</td>
</tr>
<tr>
<td></td>
<td>• Weekly by days of week: You are provided with buttons to select the days and drop down menus to select the start time in hours and minutes for the blackout window. You can select one or more days from the day buttons. The selected buttons appear with a Yellow border.</td>
</tr>
<tr>
<td></td>
<td>• Once per week: You are provided with drop down menus to select the day and start time in hours and minutes for the blackout window.</td>
</tr>
<tr>
<td></td>
<td>• Monthly by day of month: You are provided with drop down menus to select the day of the month and the start time in hours and minutes for the blackout window.</td>
</tr>
<tr>
<td></td>
<td>• Monthly by week of month: You are provided with drop down menus to select the week, the day of the week, and the start time in hours and minutes for the blackout window.</td>
</tr>
<tr>
<td>End</td>
<td>Based on the selected frequency, the following options for ending the blackout window is displayed:</td>
</tr>
<tr>
<td></td>
<td>• Weekly by days of week: You are provided with drop down menus to select the time in hours and minutes to end the blackout window.</td>
</tr>
<tr>
<td></td>
<td>• Once per week: You are provided with drop down menus to select the day of the week and time to end the blackout window.</td>
</tr>
<tr>
<td></td>
<td>• Monthly by day of month: You are provided with drop down menus to select the day of the month and the time in hours and minutes to end the blackout window.</td>
</tr>
<tr>
<td></td>
<td>• Monthly by week of month: You are provided with drop down menus to select the number of days and the time in hours and minutes to end the blackout window.</td>
</tr>
</tbody>
</table>

5. Click OK.
   The Blackout Windows tab displays the configured CMDB Sync blackout windows.
To edit an existing blackout window

You can edit an existing CMDB Sync blackout window. If the blackout is currently in progress, it is automatically cancelled when you edit it.

1. From the **Blackout Windows** tab of the CMDB Sync connection page, click **Edit** for the corresponding blackout window.
   The Edit an Existing Blackout Window dialog is displayed.
2. Edit the fields as required.
3. Click **OK**.

To delete an existing blackout window

You can delete an existing CMDB Sync blackout window.

1. From the **Blackout Windows** tab of the CMDB Sync connection page, click **Delete** for the corresponding blackout window.
2. When prompted, confirm the deletion of the blackout window by clicking **OK**.

Resyncing a CMDB connection

Occasionally the model stored in the CMDB dataset becomes out of step with the shadow copy for a CMDB connection and require resynchronization. For example, if CMDB tools have been used to modify the dataset. In this case, when updates are written to non-existent nodes, instance errors are raised. When BMC Atrium Discovery registers these instance errors, a resynchronization is recommended. In other cases such as after upgrade to version 10.2 or the creation of a new CMDB sync connection, a resynchronization is mandatory, and CMDB sync is disabled until the resynchronization is complete.

- Do not change the CMDB synchronization configuration at the same time as changing the cluster configuration.
  - Changing the CMDB sync configuration means, adding connections, removing connections, and starting or stopping a resync.
  - Changing the cluster configuration means adding members, removing members, changing coordinator, or changing fault tolerance.
The resync process runs as follows, more details are available in the CMDB Synchronization page:

1. Preparation – Prepare Resync performs the following actions, without changing any data in the CMDB:
   a. Reads identities of all CIs and relationships from the CMDB dataset.
   b. Transforms all BMC Atrium Discovery data into the local shadow copy.
   c. Compares the data read from the CMDB with the updated local shadow copy.
2. Commit – Commit Resync performs the following actions in the CMDB:
   a. Creates CIs and relationships missing from the dataset.
   b. Marks deleted any CIs or relationships present in the dataset but not in the local shadow copy.
   c. Updates all CIs and relationships to ensure their attributes match those in the local shadow copy.

To resync a CMDB connection

In the following example a new CMDB sync connection has been created, but the CMDB dataset has been used previously and is not empty. In this situation a resync is mandatory; no CMDB sync operations using this connection are possible until after the resync.

1. From the CMDB Sync page, click the Resync link. If you are on the CMDB Connection page, click the Resync tab.
   The CMDB Connection page is displayed with the Resync tab active.

2. Click Start Resync. The following steps are performed and the screen refreshed to show progress:
   a. Reads identities of all CIs and relationships from the CMDB dataset.
   b. Transforms all BMC Atrium Discovery data into the local shadow copy.
2. c. Compares the data read from the CMDB with the updated local shadow copy. At this stage, no changes have been made to the CMDB. Here you can review the changes that are required to resync. Check that you see what you expect to see. For example, if you intend to resync a new connection to an empty CMDB dataset, then many updates or deletes may indicate that you have specified the wrong dataset.

3. Click **Commit Resync**.
   a. Creates CIs and relationships missing from the dataset.
   b. Marks deleted any CIs or relationships present in the dataset but not in the local shadow copy.
   c. Updates all CIs and relationships to ensure their attributes match those in the local shadow copy.

When this is complete, the resync is reported as complete.
3. When the resync is complete, you can resume CMDB sync by selecting Resume Sync from the Actions menu.

Managing CMDB synchronization

The CMDB Sync page displays all configured CMDB sync connections, and provides the following controls:

- **Global actions** – to perform tasks on the CMDB process as an entity, effective over all connections.
- **Individual actions** – to perform tasks on individual CMDB sync connections.

4. Do not change the CMDB synchronization configuration at the same time as changing the cluster configuration.

   - Changing the CMDB sync configuration means, adding connections, removing connections, and starting or stopping a resync.
   - Changing the cluster configuration means adding members, removing members, moving the coordinator, or changing fault tolerance.
Each connection pane is color-coded to show its state, and an informational message on the state is included in the colored section. The color codes and their meaning are:

- **Green** – CMDB connection OK.
- **Yellow** – the server has errors.
- **Blue** – the connection has been manually paused or a blackout window is in operation.
- **Red** – the server is unreachable, credentials are incorrect, or an unexpected connection error exists.
- **Grey** – connection disabled.

During synchronization, the operation can fail with the following type of message:

```
ERROR (8760): Cannot establish a network connection to the AR System Plug-In server; [cmdb hostname] (9555) BMC.FILTERAPI.NORM. ENGINE : RPC: Miscellaneous tli error - System error (Connection refused)
```

This can be caused by Java being upgraded on the CMDB target, without a corresponding update to the following file to point to the new `java/bin` directory:

```
C:\Program Files\BMC Software\ARSystem\Conf\armonitor.cfg
```

Each connection pane also provides information on the connection:

- **Name** – also a link to the connection page.
- **Continuous sync status** – on or off.
- **Network Address** – the address of the BMC Atrium CMDB server.
- **Dataset** – the CMDB dataset that the connection is syncing into.
- **Device nodes** – counts of nodes inserted, updated, and deleted. May be reset using the reset counters action menu item.
- **Data model** – the data model in use for the connection.
- **Currently processing** – whether any nodes are currently being synced.
- **Devices in queue** – the number of devices that are currently being queued.
Performing tasks on all CMDB sync connections

The following tasks are performed on all CMDB sync connections:

**Enabling and disabling continuous sync**

Synchronization can be configured to run in a *continuous* mode, meaning that as soon as BMC Atrium Discovery completes its scan of a device, the data corresponding to it is queued for synchronization with the CMDB. Alternatively, synchronization can be launched manually for one or more devices from within the browsing interface.

- Global Actions menu => **Continuous Sync On**
- Global Actions menu => **Continuous Sync Off**

**Pausing and resuming sync**

Synchronization can be paused, meaning that newly discovered or updated nodes, and deletions, are still added to the sync queues, but are not processed until synchronization is resumed.

- Global Actions menu => **Pause Sync**
- Global Actions menu => **Resume Sync**

**Enabling and disabling sync**

Disabling sync means that newly discovered or updated nodes, and deletions, will no longer be added to the sync queues.

- Global Actions menu => **Disable Sync**
- Global Actions menu => **Enable Sync**
Performing tasks on individual CMDB sync connections

The following tasks are available from the Actions drop down for each CMDB sync connection:

- Actions drop down => **Edit**
  Edit the connection details. Opens the Manage CMDB Connection page with the Details tab showing. This can also be accessed by clicking the connection name link.

- Actions drop down => **Delete**
  Delete the connection. You are prompted for confirmation

- Actions drop down => **Continuous Sync On/Continuous Sync Off**
  Turn continuous sync on or off.

- Actions drop down => **Pause Sync/Resume Sync**
  Pause or resume sync.

- Actions drop down => **Enable Sync/Disable Sync**
  Enable or disable sync.

- Actions drop down => **Reset Counters**
  Reset the counts of nodes inserted, uploaded, and deleted for this connection.

Batch and individual synchronization

Nodes can be synchronized into BMC Atrium CMDB as a batch. From a list view in the UI, select one or more nodes, then choose CMDB Sync from the Actions menu. The submenu enables you to choose to **Sync to all connections**, or any one of the configured connections.

The menu items appear for any root node kind (Host, MFPart, NetworkDevice, Printer, SNMPManagedDevice, and Storage System).

To see the progress of the synchronization, you can follow the link in the notification to the status page.

A node is only added to the synchronization queue if it is not already queued.

To synchronize an individual device

From a node view page for a root node kind, you can choose CMDB Sync from the Actions menu. The submenu enables you to sync the device to all connections, or any one of the configured connections.
To synchronize deletion of root nodes

When root nodes are deleted in the BMC Atrium Discovery datastore, the associated CIs will be deleted in the CMDB. In continuous synchronization mode, deletions are automatically queued. To synchronize deletions using batch or individual synchronization, view the destroyed node(s), and then choose **Sync with CMDB** from the **Actions** menu.

There are many ways to find destroyed nodes. One simple way is to choose **Advanced Search**, select the **Include Destroyed** check box, and then choose the node kind (for example, Host) that you want to search for.

**Default CDM Mapping**

**Root node kind mappings**

Each root node kind in the Discovery model is mapped to the Common Data Model (CDM) as follows:

- CDM Mapping for Host
  - Extending the CDM mapping
- CDM Mapping for MFPart
- CDM Mapping for Network Device
  - CDM Mapping for Load Balancer nodes
- CDM Mapping for Printer
- CDM Mapping for SNMP Managed Device
- CDM Mapping for Storage

⚠️ **Restart the Tideway service after making changes to CDM**

If any changes are made to the CDM in the CMDB, for example, adding attributes, you cannot sync to those attributes until the tideway service has been restarted. On restart of the tideway service, the CDM is read and all customized classes and attributes are available to CMDB sync.

**Extending and modifying the standard mappings**

If you need to extend or modify the standard mappings, it is best to do so with extension mappings, rather than by editing the standard mappings. Extension mappings are able to create new CIs and relationships, set additional CI attributes, and change the standard values of existing CI attributes. The standard mappings should only be edited as a last resort, if the mapped structure needs to be different. See the **Pattern templates** for examples.
ADDMIntegrationId
So that Discovery can correctly maintain the CIs it creates in its dataset, it stores a unique \textit{key} on every CI, in the ADDMIntegrationId attribute. These keys are sometimes directly populated from the key on a corresponding node in the Discovery datastore, but in other situations they are constructed by using rules that are appropriate for the mapping structure. See the syncmapping definitions for details of how keys are populated.

Company attribute
If the CMDB is configured for multitenancy, all CIs can have a \texttt{Company} attribute set appropriately. See \texttt{Multitenancy} for details.

Impact relationships
One of the main features of BMC Atrium CMDB is to indicate the way that one CI impacts another. Versions of BMC Atrium CMDB prior to 7.6.03 represent impact using a specific relationship, \texttt{BMC_Impact}. With those CMDB versions, Discovery is responsible for creating and maintaining the \texttt{BMC_Impact} relationships. In later CMDB versions, the \texttt{BMC_Impact} relationship has been deprecated, and impact is now indicated with the \texttt{HasImpact} attribute on all other relationship classes.

BMC Service Impact Manager (SIM) version 7.4 is incompatible with the new representation of impact as an attribute, so if SIM is used with CMDB version 7.6.03 or later, Discovery must be configured with the "CMDB 7.6.03 and later with Impact relationships" data model. In that case, Discovery will ask the CMDB to create \texttt{BMC_Impact} relationships, but the deprecation mechanism will translate them into \texttt{BMC_BaseRelationship} instances with the \texttt{Name} "ImpactOnly". If you view such a relationship in Atrium Explorer, it will display as a \texttt{BMC_BaseRelationship} with \texttt{HasImpact} set to "Yes". However, the deprecation mechanism means that it is still possible to find the relationships as \texttt{BMC_Impact} by using the CMDB API or by using the \texttt{BMC.CORE:BMC_Impact} form. This complicated situation \textit{only} applies to BMC Service Impact Manager version 7.4 and earlier, not to the SIM functionality built into later versions of BMC Proactive Performance Manager.

CDM Mapping for Host
This topic provides information for CDM mapping for Host nodes.

- \texttt{BMC_ComputerSystem}
- \texttt{BMC_OperatingSystem}
- \texttt{BMC_Processor}
- \texttt{BMC_NetworkPort}, \texttt{BMC_IPEndpoint} and \texttt{BMC_LANEndpoint}
- \texttt{BMC_IPConnectivitySubnet}
- \texttt{BMC_Cluster}
- \texttt{BMC_VirtualSystemEnabler} and virtualization
- \texttt{BMC_HardwareSystemComponent}
- \texttt{BMC_ComputerSystem (Host Container)}
Host nodes in the Discovery model are mapped to the Common Data Model (CDM) as shown in this automatically-generated diagram:

This diagram illustrates how the Host nodes in the Discovery model are mapped to the Common Data Model.

The CIs created are described in this topic.

**Data models**

### CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with `datamodel` declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td></td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
</tbody>
</table>
### Differences from previous versions of BMC Atrium Discovery

- Except where noted below, the model populated by BMC Atrium Discovery version 9.0 is the same as that populated by earlier versions.
- Starting from version 10.0, in the model populated by BMC Atrium Discovery, the `ram` attribute is now used for storing the physical memory volume and the new `logical_ram` attribute is added for the reported logical memory volume.

### BMC_ComputerSystem

The root Host node is mapped to a root BMC_ComputerSystem CI with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The fully qualified name of the host if available, otherwise the unqualified name; if no names are available, the IP address</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;DNS&quot;, &quot;HostName&quot; or &quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Host name attribute</td>
</tr>
<tr>
<td>Description</td>
<td>The host name and the fully qualified domain name separated by a colon</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>A single value corresponding to Server, Desktop, or Laptop</td>
</tr>
<tr>
<td>Domain</td>
<td>Host <code>dns_domain</code> — the DNS domain of the host</td>
</tr>
<tr>
<td>HostName</td>
<td>Host <code>hostname</code> — the hostname reported by the host</td>
</tr>
<tr>
<td>isVirtual</td>
<td>Yes if the host is known to be virtual (or partitioned hardware); null if not</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>Host <code>last_update_success</code> — the date and time the host was last scanned</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Host <code>vendor</code> — the manufacturer of the host</td>
</tr>
<tr>
<td>Model</td>
<td>Host <code>model</code></td>
</tr>
</tbody>
</table>
1. **PrimaryCapability**
   - Details: Server, Desktop or Laptop

2. **SerialNumber**
   - Details: Host serial

3. **SystemType**
   - Details: Enumeration value corresponding to the system type

4. **TokenId**
   - Details: See **TokenId rules** below

   **PartitionId**
   - Details: Host partition_id – the identifier of a partition host (new in CMDB 8.1) - from TKU 2014-Mar-1 onwards

   **TotalPhysicalMemory**
   - Details: Host ram — total Host ram

   **VirtualSystemType**
   - Details: Enumeration value corresponding to the type of virtual machine

   **Workgroup**
   - Details: Host workgroup — the Windows workgroup

   **Category**
   - Details: "Hardware"

   **Type**
   - Details: "Processing unit"

   **Item**
   - Details: "Server", "Desktop" or "Laptop"

**TokenId rules**

**TokenId** is an attribute that in some circumstances aids reconciliation of CIs populated by multiple data sources. The following describes how discovery sets TokenId for **BMC_ComputerSystem**.

For most hosts, **TokenId** is of the form **hostname**: **DNS domain name.**

For some virtual hosts, **TokenId** contains a UUID:

1. For VMware, **TokenId** is of the form "VI-UUID: ABCD-EF-GH-IJ-KLMNOP". Where each letter represents a hexadecimal digit.
2. For Hyper-V, **TokenId** is of the form "HYPERV-ID: ABCD-EF-GH-IJ-KLMNOP". With Hyper-V, the UUID is only available on the physical machine, so TokenId is only set for virtual machines that have been successfully linked to their hosting physical machines.
3. For Xen (including Oracle VM), **TokenId** is of the form "XEN-ID: ABCD-EF-GH-IJ-KLMNOP".
4. For KVM (including RedHat Enterprise Virtualization), **TokenId** is of the form "KVM-ID: ABCD-EF-GH-IJ-KLMNOP".

**BMC_OperatingSystem**

The root Host is also mapped to a single **BMC_OperatingSystem** CI, with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Host os_type</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;OSName&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>&quot;OS type version on hostname&quot;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Host os</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Host os_vendor</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Host os_version</td>
</tr>
<tr>
<td>Model</td>
<td>Host os_type</td>
</tr>
<tr>
<td>OSType</td>
<td>Enumeration value for OS type</td>
</tr>
<tr>
<td>OSProductSuite</td>
<td>Enumeration value for OS product suite</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Host service_pack</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Host os_version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Operating System&quot;</td>
</tr>
</tbody>
</table>

**Operating system relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMOS</td>
<td>BMC_ComputerSystem</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_OperatingSystem</td>
<td>BMC_ComputerSystem</td>
</tr>
</tbody>
</table>

**BMC_Processor**

The root Host is mapped to a number of BMC_Processor CIs.

- For physical machines, the number of CIs corresponds to the number of physical CPU packages present in the machine.
- For virtual machines, the number of CIs corresponds to the number of logical CPUs the OS is scheduling across.

In some circumstances, it is not possible to discover the number of physical CPU packages in a physical machine. In such cases, no BMC_Processor CIs are created for the machine.

All the BMC_Processor CIs for a host are normally identical to each other. New in BMC Atrium Discovery 8.3, they can be different in cases that a physical machine has more than one type of CPU.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>&quot;CPU\text{index}&quot; (for example, &quot;CPU0&quot;, &quot;CPU1&quot;, and so on.)</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;ProcessorName&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Host processor_type</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the processor and its index</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>isVirtual</td>
<td>True if the Host is virtual; not set if physical. (Not present in CMDB 7.5.)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Manufacturer of the processor. (New in ADDM 8.3.)</td>
</tr>
<tr>
<td>MaxClockSpeed</td>
<td>Host processor_speed</td>
</tr>
<tr>
<td>Model</td>
<td>Host processor_type</td>
</tr>
<tr>
<td>NumberOfCores</td>
<td>Host cores_per_processor (only set for physical machines)</td>
</tr>
<tr>
<td>NumberOfLogicalProcessors</td>
<td>Host num_logical_processors / num_processors (only set for physical machines)</td>
</tr>
<tr>
<td>OtherProcessorFamilyDescription</td>
<td>Host processor_type (only set if ProcessorFamily is &quot;Other&quot;)</td>
</tr>
<tr>
<td>ProcessorArchitecture</td>
<td>Enumeration representing the architecture</td>
</tr>
<tr>
<td>ProcessorFamily</td>
<td>Enumeration representing the processor family</td>
</tr>
<tr>
<td>ProcessorStatus</td>
<td>1 — CPU enabled</td>
</tr>
<tr>
<td>ProcessorType</td>
<td>2 — Central processor</td>
</tr>
<tr>
<td>UpgradeMethod</td>
<td>1 — Unknown</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Component&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;CPU&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Difference from BMC Atrium Discovery 8.1**

BMC Discovery version 8.1 always created one single `BMC_Processor CI`, regardless of the circumstances.

### Processor relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Processor</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort, BMC_IPEndpoint and BMC_LANEndpoint**

Each Network Interface node connected to the root Host in the Discovery model is mapped to a `BMC_NetworkPort CI`, and its MAC address is mapped to a `BMC_LANEndpoint CI`. Each of its associated IP addresses are mapped to `BMC_IPEndpoint CIs`.

Where Discovery has been able to connect a Network Interface node to a Network Device node (via associated port nodes), a `BMC_Dependency relationship` is created between the two `BMC_ComputerSystem CIs` representing the host and the network device.
### BMC_NetworkPort

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>Description</td>
<td>NetworkInterface name”</td>
</tr>
<tr>
<td>AutoSense</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>FullDuplex</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>NetworkInterface manufacturer</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>PortType</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>Speed</td>
<td>NetworkInterface raw_speed</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Network interface card&quot;</td>
</tr>
</tbody>
</table>

Each Fibre Channel HBA card node is mapped to the BMC_NetworkPort CI with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>FibreChannelHBA description and FibreChannelPort wwpn or just FibreChannelPort wwpn</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>FibreChannelPort wwpn</td>
</tr>
<tr>
<td>Description</td>
<td>FibreChannelHBA description and FibreChannelPort wwpn or just FibreChannelPort wwpn</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>Fibre Channel (4)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>FibreChannelHBA vendor</td>
</tr>
<tr>
<td>Model</td>
<td>FibreChannelHBA model</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>FibreChannelPort wwpn</td>
</tr>
<tr>
<td>PortType</td>
<td>Fibre Channel (6)</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>FibreChannelHBA serial</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Fibre Channel HBA card&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**
BMC_NetworkPort was not created in versions prior to 8.3.

**BMC_LANEndpoint**

MAC addresses in the Discovery data model are stored in the conventional form with colons separating each pair of hexadecimal digits; in the CDM, MAC addresses are stored as just the hexadecimal digits, with no separating colons.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;MAC&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;MAC address on hostname&quot;</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>Ethernet (14)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;MAC Address&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

Prior to version 8.3, Category was set to "Miscellaneous" instead of "Network," and Address was set to the "IP address", not the MAC address.

**BMC_IPEndpoint**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>Description</td>
<td>IP Address name</td>
</tr>
<tr>
<td>Address</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>DNSHostName</td>
<td>IP Address fqdns (first item in list)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>IPv4 (2) or IPv6 (3)</td>
</tr>
</tbody>
</table>
### Differences from earlier BMC Atrium Discovery versions

Prior to version 8.3, Category was set to "Miscellaneous" instead of "Network".

### NetworkPort and Endpoint relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_NetworkPort</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>BINDSTO</td>
<td>BMC_LANEndpoint</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>NETWORKLINK</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
</tbody>
</table>

### BMC_IPConnectivitySubnet

**Discovery** Subnet is mapped to BMC_IPConnectivitySubnet:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Description</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IP Address prefix_length</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>SubnetNumber</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Subnet&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;TCP/IP&quot;</td>
</tr>
</tbody>
</table>

### Subnet relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_InIPSubnet</td>
<td>INIPSUBNET</td>
<td>BMC_IPConnectivitySubnet</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>

### BMC_Cluster

Host Cluster nodes are mapped to **BMC_Cluster**:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Cluster name</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;ClusterName&quot;</td>
</tr>
<tr>
<td>Description</td>
<td>Cluster name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Cluster name</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Unknown&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Unknown&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;BMC Discovered&quot;</td>
</tr>
</tbody>
</table>

### Cluster relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>CLUSTEREDSYSTEM</td>
<td>BMC_Cluster</td>
<td>BMC_ComputerSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Cluster</td>
</tr>
</tbody>
</table>

### BMC_VirtualSystemEnabler and virtualization

When a physical machine hosts one or more virtual machines, the Discovery model represents it as a number of Software Instance nodes related to the physical Host, one Software Instance per virtual host. In the CDM, the BMC_ComputerSystem corresponding to the physical machine has a BMC_HostedSystemComponents relationship to a single BMC_VirtualSystemEnabler CI. (Strictly speaking, it is one per technology, so a physical machine running more than one virtualization technology could have more than one BMC_VirtualSystemEnabler CI.)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Instance name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>&quot;VM type on hostname&quot;</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>&quot;VM type on hostname&quot;</td>
</tr>
<tr>
<td>BuildNumber</td>
<td>Software Instance build</td>
</tr>
<tr>
<td>EnablerType</td>
<td>Enumeration representing the type of virtualization technology</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Publisher from the Pattern maintaining the Software Instance</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Software Instance product_version</td>
</tr>
<tr>
<td>Model</td>
<td>Software Instance type</td>
</tr>
<tr>
<td>PatchNumber</td>
<td>Software Instance patch</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Software Instance service_pack</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Virtualization OS&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

Prior to version 8.3, Name was set to the BMC_ComputerSystem Name.

### Virtualization relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>HOSTEDSYSTEMCOMPONENTS</td>
<td>BMC_ComputerSystem (physical)</td>
<td>BMC_VirtualSystemEnabler</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>VIRTUALSYSTEMOS</td>
<td>BMC_VirtualSystemEnabler</td>
<td>BMC_ComputerSystem (virtual)</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>HOSTEDVIRTUALSYSTEM</td>
<td>BMC_ComputerSystem (physical)</td>
<td>BMC_ComputerSystem (virtual)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (physical)</td>
<td>BMC_VirtualSystemEnabler</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_VirtualSystemEnabler</td>
<td>BMC_ComputerSystem (virtual)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (physical)</td>
<td>BMC_ComputerSystem (virtual)</td>
</tr>
</tbody>
</table>

**BMC_HardwareSystemComponent.**

Any Management Controller node with a direct Management relationship to the root Host in the Discovery model is mapped to a BMC_HardwareSystemComponent CI.
Any ManagementController for a HostContainer containing the root Host in the Discovery model is mapped to BMC_HardwareSystemComponent.

⚠️ There is no way to map the ManagementController's IP information to the CDM.

**BMC_HardwareSystemComponent**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Management controller name.</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;HostName&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Management controller name.</td>
</tr>
<tr>
<td>Description</td>
<td>Vendor name, controller model and type. With serial if known.</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Management controller vendor name.</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Management controller version number.</td>
</tr>
<tr>
<td>Model</td>
<td>Management controller model.</td>
</tr>
<tr>
<td>PowerManagementSupported</td>
<td>Yes (0) or null if not known.</td>
</tr>
<tr>
<td>Company</td>
<td>Host Company.</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Hardware&quot;</td>
</tr>
</tbody>
</table>
| Item             | *Management controller* where managed element is a directly connected host.  
"Chassis manager" where the managed element is a chassis. |

**Hardware system component relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_HardwareSystemComponent</td>
</tr>
</tbody>
</table>

**BMC_ComputerSystem (Host Container)**

HostContainer nodes are mapped to BMC_ComputerSystem representing the partitioned container.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Host Container name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Host Container name</td>
</tr>
<tr>
<td>Description</td>
<td>Host Container name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>&quot;Blade System&quot; (48) or &quot;Other&quot; (2)</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>Host last_update_success</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Host Container vendor</td>
</tr>
</tbody>
</table>
### Host containment relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>SYSTEMPARTITION</td>
<td>BMC_ComputerSystem (container)</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (container)</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
</tbody>
</table>

### BMC_SoftwareServer and BMC_ApplicationSystem

Each SoftwareInstance in the Discovery model is mapped to either a BMC_SoftwareServer CI or a BMC_ApplicationSystem CI:

- SoftwareInstance nodes with DDD triggers are directly running on a single host, and are mapped to BMC_SoftwareServer.
- SoftwareInstance nodes composed of other SoftwareInstance nodes are mapped to BMC_ApplicationSystem.

SoftwareInstance nodes are also mapped (in an indirect manner) to BMC_Product.

Except as noted, the attributes set are the same in BMC_SoftwareServer and BMC_ApplicationSystem.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Instance name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Software Instance name</td>
</tr>
<tr>
<td>Description</td>
<td>Software Instance name</td>
</tr>
<tr>
<td>BuildNumber</td>
<td>Software Instance build</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Publisher specified in the maintaining Pattern or on the Software Instance</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Software Instance product_version</td>
</tr>
<tr>
<td>Model</td>
<td>Software Instance type</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>OtherSoftwareServerType</td>
<td>Software Instance type (Only in BMC_SoftwareServer)</td>
</tr>
<tr>
<td>PatchNumber</td>
<td>Software Instance patch</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Software Instance service_pack</td>
</tr>
<tr>
<td>SoftwareServerType</td>
<td>Enumeration representing the type of SoftwareServer. Often 0 meaning &quot;Other&quot;. (Only in BMC_SoftwareServer)</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>Product category from the maintaining Pattern</td>
</tr>
</tbody>
</table>

### Item values

The Item attribute is populated from the Pattern that is maintaining the Software Instance. To obtain a list of all the possible Item values, perform the following query in the Discovery Generic Query page:

```plaintext
search Pattern show categories processwith countUnique(0)
```

### Differences from earlier BMC Atrium Discovery versions

Prior to version 8.3, Name was set to "hostname: type: name".

Version 8.2 mapped all SoftwareInstance nodes to BMC_SoftwareServer. Version 8.1 mapped those SoftwareInstances containing other SoftwareInstances to BMC_ApplicationInfrastructure.

Version 8.1 always hard-coded Item to "BMC_Discovered".

### BMC_SoftwareServer and BMC_ApplicationSystem relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMCOMPUTER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMCOMPUTER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_ApplicationSystem</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_ApplicationSystem</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMDEPENDENCY</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
As shown in the table, BMC_SoftwareServer and BMC_ApplicationSystem CIs can be related to each other with a BMC_Dependency relationship. This is mapped from both Dependency and Communication relationships between the SoftwareInstance nodes in the Discovery model.

BMC_SoftwareServer and BMC_ApplicationSystem CIs also have relationships to BMC_Product, shown in the table below.

**BMC_Product**

BMC_Product represents the installed aspects of a product. BMC Atrium Discovery uses SoftwareInstance nodes and associated metadata to provide an accurate picture of the installed products, regardless of how the products are installed. The mapping groups the SoftwareInstance nodes according to the product information on the maintaining patterns, and maintains one BMC_Product CI for each unique product version on the Host. SoftwareInstance nodes with different types can belong to a single product. So, for example, a server running four Oracle database instances and one Oracle TNS Listener would have five SoftwareInstance nodes; those five SoftwareInstances would map to a single Oracle Database BMC_Product CI.
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>&quot;Software Application/System&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>Product category from the maintaining Pattern</td>
</tr>
</tbody>
</table>

#### Item values

The Item attribute is populated from the Pattern that is maintaining the Software Instance. To obtain a list of all the possible Item values, perform the following query in the Discovery Generic Query page:

```sql
search Pattern show categories processwith countUnique(0)
```

#### Difference from BMC Atrium Discovery 8.1

BMC Atrium Discovery version 8.1 always hard-coded Item to "BMC Discovered".

### Product relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>INSTALLEDSOFTWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Product</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMPRODUCT</td>
<td>BMC_Product</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMPRODUCT</td>
<td>BMC_Product</td>
<td>BMC_ApplicationSystem</td>
</tr>
</tbody>
</table>

**BMC_ApplicationService**

SoftwareComponent is mapped to BMC_ApplicationService.

### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Component name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Software Component name</td>
</tr>
<tr>
<td>Description</td>
<td>Software Component name</td>
</tr>
<tr>
<td>Model</td>
<td>Software Component type</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Component version</td>
</tr>
<tr>
<td>ApplicationServiceType</td>
<td>Enumeration value representing the service type</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application Service&quot;</td>
</tr>
</tbody>
</table>
**Application service relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_ApplicationSystemServices</td>
<td>APPLICATIONSYSTEMSERVICES</td>
<td>BMC_SoftwareServer</td>
<td>BMC_ApplicationService</td>
</tr>
</tbody>
</table>

**BMC_DataBase**

DatabaseDetail nodes representing logical databases are mapped to BMC_DataBase:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>DatabaseDetail name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>DatabaseDetail instance</td>
</tr>
<tr>
<td>Description</td>
<td>DatabaseDetail name</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Software Instance vendor</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Software Instance product_version</td>
</tr>
<tr>
<td>Model</td>
<td>DatabaseDetail type</td>
</tr>
<tr>
<td>PatchNumber</td>
<td>Software Instance patch</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Software Instance service_pack</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Miscellaneous&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Instance&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Database&quot;</td>
</tr>
</tbody>
</table>

**BMC_DataBase relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>MANAGEDDATABASE</td>
<td>BMC_SoftwareServer</td>
<td>BMC_DataBase</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_DataBase</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

BMC_DataBase was not created in version 8.2.
**BMC_Application**

BusinessApplicationInstance is mapped to BMC_Application. Contained SoftwareInstance and BusinessApplicationInstance nodes are mapped to corresponding BMC_Component relationships.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>Description</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Business Application Instance product_version</td>
</tr>
<tr>
<td>Model</td>
<td>Business Application Instance type</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Business Application Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Application Platform&quot;</td>
</tr>
</tbody>
</table>

**Application relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMCOMPUTER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_Application</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_Application</td>
<td>BMC_ApplicationSystem</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_Application (containing)</td>
<td>BMC_Application (contained)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ApplicationSystem</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_Application (containing)</td>
<td>BMC_Application (containing)</td>
</tr>
</tbody>
</table>

**BMC_ConcreteCollection**

FunctionalComponent nodes representing CAM Functional Components are mapped to BMC_ConcreteCollection.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Functional Component name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Functional Component name</td>
</tr>
<tr>
<td>Description</td>
<td>Functional Component name</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Functional Component type</td>
</tr>
<tr>
<td>Company</td>
<td>Host Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Functional Component&quot;</td>
</tr>
</tbody>
</table>

#### Functional Component relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_SoftwareServer First-order SI.</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_ApplicationSystem Second-order SI.</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_DataBase</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_ApplicationService</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>FUNCTIONALCOMPONENTINAPPLICATION</td>
<td>BMC_Application</td>
<td>BMC_ConcreteCollection</td>
</tr>
</tbody>
</table>

The relationship between **BMC_Application** and **BMC_ConcreteCollection** is used for Functional Components reached from any of the following node kinds:

- SoftwareInstance
- DatabaseDetail
- BAI
- SoftwareComponent

#### Extending the CDM mapping

The following example shows how to extend the default CDM mapping for a host to create **BMC_PhysicalLocation** CIs from location information determined in BMC Atrium Discovery and relate these to **BMC_ComputerSystem** CIs.

The first pattern is used to determine the location of the host. Taken from Using TPL to enrich discovered data:

**Using a table in a pattern to relate subnets to locations**

Subnets can be used to identify locations of hosts. You can extend the `template_host_location` pattern to map subnets to locations. The following TPL snippet shows how you could use a table to map the subnet to hosts. You could also hard code a mapping of hostnames to locations if that information is available. See TPL Guide for more information on tables.
table SubnetLocations 1.0

"101.10.10.0/24" -> "London Victoria";
"101.10.11.0/24" -> "London Egham";
"137.72.94.0/24" -> "Houston";
"137.72.95.0/24" -> "Dallas";
default -> "unknown";
end table;

The following TPL snippet shows the search from the host to associated subnet or subnets.

subnet_list := search(in host traverse DeviceWithAddress:DeviceAddress:IPv4Address:IPAddress
traverse DeviceOnSubnet:DeviceSubnet:Subnet:Subnet);
location_id := '';
list_size := size(subnet_list);
log.debug("Subnet_2_Locations: subnet_list size is %list_size%);
for subnet_node in subnet_list do
    log.debug("Subnet_2_Locations: Current subnet_node is %subnet_node.ip_address_range%";)
    // if subnet_node is "None" get next node otherwise we can use this one
    if '%subnet_node.ip_address_range%' <> "None" then
        location_id:= '%subnet_node.ip_address_range%';
        break;
    end if;
end for;

This TPL snippet uses the table to look up the location from the subnet.

// Look up the location name from the table
location_name := SubnetLocations[location_id];

Now you can create or update the relationship between the host and location node. The following code snippet uses the model.uniquerel function to create or update the relationship.

// Relate host to location. Using "uniquerel" rather than "rel" means that
// any existing Location relationships between this Host (the first
// parameter) and any Location nodes other than the one given (the second
// parameter) are removed. If a host has changed location, this keeps the
// model up-to-date.
log.info("Subnet_2_Locations: model.uniquerel.Location %host% %location%";
model.uniquerel.Location(
```plaintext
ElementInLocation := host,
Location := location
```

The host is now related to the location node which represents its physical location.

The second pattern is the template_cmdb_location template which is available in the Pattern Templates section of the Pattern Management page. See Pattern templates for more information on the template patterns supplied with BMC Atrium Discovery. The syncmapping is described in syncmapping the TPL Guide.

The template_cmdb_location syncmapping extends the root mapping for Host nodes. The mapping section specifies the extension to the root mapping using the from keyword. The traversal finds related location nodes, and the mapping from phys_loc to BMC_PhysicalLocation is defined.

The name defined by the traversal can only be used in a for each expression; it cannot be used in any other context.

```plaintext
mapping from Host_ComputerSystem.host as host
  traverse ElementInLocation:Location:Location:Location as location
    phys_loc -> BMC_PhysicalLocation;
  end traverse;
end mapping;
```

The body of the template_cmdb_location pattern loops through each location reached by the traversal, and transforms the subgraph of data in the Discovery model into a subgraph of CIs in the target CMDB model. This is copied across and then the relationship is created between the BMC_ComputerSystem and the new BMC_PhysicalLocation in the CMDB.

```plaintext
body
  computersystem := Host_ComputerSystem.computersystem;
  for each location do
    // ADDM Location nodes created by patterns have a key
    // attribute, but those created in the UI do not. We use
    // the name attribute if the key is not present.
    if location.key then
      key := location.key;
    else
      key := location.name;
```
end if;

// Specify the BMC_PhysicalLocation CI. You may wish to
// add more attributes to the ADDM Location node and copy
// them across here.
phys_loc := sync.shared.BMC_PhysicalLocation(
    key := key,
    Name := location.name,
    ShortDescription := location.name,
    Description := location.description,
    Address := location.address,
    Category := "Reference Configuration",
    Type := "Facilities",
    Item := "Space Management"
);

// Relate the BMC_ComputerSystem to the new BMC_PhysicalLocation
sync.rel.BMC_ElementLocation(
    Source := computersystem,
    Destination := phys_loc,
    Name := "ELEMENTLOCATION"
);

end for;
end body;

CDM Mapping for MFPart

This topic provides information about CDM mapping for MFPart nodes.

- BMC_ComputerSystem
- BMC_OperatingSystem
- BMC_Mainframe
- BMC_VirtualSystemEnabler
- BMC_Sysplex / BMC_Cluster
- BMC_MFCouplingFacility
- BMC_StorageSubsystem
- BMC_DiskDrive / BMC_TapeDrive
- BMC_MFSoftwareServer / BMC_SoftwareServer
- BMC_ApplicationService
- BMC_Application
- BMC_DataBase
- BMC_SystemResource
- BMC_Transaction
- BMC_ConcreteCollection
MFPart nodes in the Discovery model are mapped to the Common Data Model as shown in this automatically-generated diagram:

![Diagram showing MFPart nodes mapping to Common Data Model](image)

The CIs created are described in this topic.

**Data models**

### CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- **BMC_Impact** relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name “ImpactOnly” are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>

![Warning symbol]
Differences from previous versions of BMC Atrium Discovery

Except where noted below, the model populated by BMC Atrium Discovery version 9.0 is the same as that populated by earlier versions.

BMC_ComputerSystem

The MFPart node is used to model a logical server within the mainframe. The root MFPart node is mapped to a root BMC_ComputerSystem CI with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>MFPart key attribute</td>
</tr>
<tr>
<td>Name</td>
<td>MFPart key attribute</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>MFPart name attribute</td>
</tr>
<tr>
<td>Description</td>
<td>MFPart description attribute</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>Server</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>isVirtual</td>
<td>Yes if either MFPart virtual or partition is set; null otherwise</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>MFPart last_update_success — the date and time the MFPart was last scanned</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>MFPart vendor — the manufacturer of the mainframe</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>MFPart model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Server</td>
</tr>
<tr>
<td>VirtualSystemType</td>
<td>LPAR or VM/VM Guest</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Processing Unit&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Server&quot;</td>
</tr>
</tbody>
</table>

BMC_OperatingSystem

The root MFPart is also mapped to a single BMC_OperatingSystem CI, with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>MFPart __mfpart_id</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>OS type followed by OS serial number</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;OS-__mfpart_id&quot;</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>MFPart vendor</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>MFPart os_type — &quot;z/OS&quot; or &quot;z/VM OS&quot;</td>
</tr>
<tr>
<td>OSType</td>
<td>z/OS or z/VM</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>MFPart os_serial</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>MFPart os_version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Standard OS&quot;</td>
</tr>
</tbody>
</table>

Operating system relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMOS</td>
<td>BMC_ComputerSystem</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_OperatingSystem</td>
<td>BMC_ComputerSystem</td>
</tr>
</tbody>
</table>

**BMC_Mainframe**

An MFPart corresponding to a Native LPAR is directly connected to a Mainframe node. The Mainframe is mapped to a BMC_Mainframe CI as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mainframe key</td>
</tr>
<tr>
<td>Description</td>
<td>Mainframe description</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Mainframe name</td>
</tr>
<tr>
<td>AvailableCPs</td>
<td>Mainframe available_cps</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>Mainframe</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>InstalledCPUs</td>
<td>Mainframe num_processors</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Mainframe vendor</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>Mainframe mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>Mainframe model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Mainframe</td>
</tr>
<tr>
<td>Category</td>
<td>Hardware</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>Type</td>
<td>Processing Unit</td>
</tr>
<tr>
<td>Item</td>
<td>Mainframe</td>
</tr>
</tbody>
</table>

### Mainframe relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>SYSTEMPARTITION</td>
<td>BMC_Mainframe</td>
<td>BMC_ComputerSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_Mainframe</td>
<td>BMC_ComputerSystem</td>
</tr>
</tbody>
</table>

### BMC_VirtualSystemEnabler

For Native LPARs, the MFPart and Mainframe nodes are mapped to a BMC_VirtualSystemEnabler CI as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mainframe key with suffix “-VSE”</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>MFPart part_type</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>Description</td>
<td>MFPart part_type followed by Mainframe key</td>
</tr>
<tr>
<td>EnablerType</td>
<td>MFPart __cdm_enabler_type</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>MFPart vendor</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>Mainframe mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>MFPart part_type</td>
</tr>
<tr>
<td>Category</td>
<td>“Software”</td>
</tr>
<tr>
<td>Type</td>
<td>“Operating System Software”</td>
</tr>
<tr>
<td>Item</td>
<td>“Virtualization OS”</td>
</tr>
</tbody>
</table>

For zVM LPARs, the Discovery model represents the virtualization as a Software Instance node connected to the containing Native LPAR. That Software Instance is mapped to a BMC_VirtualSystemEnabler as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Instance __cdm_component_alias</td>
</tr>
<tr>
<td>NameFormat</td>
<td>“Mainframe”</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Software Instance vm_id</td>
</tr>
<tr>
<td>Description</td>
<td>“z/VM-” followed by Software Instance __cdm_component_alias</td>
</tr>
<tr>
<td>BuildNumber</td>
<td>Software Instance build</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>EnablerType</td>
<td>Software Instance cdm_vm_enabler_type</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>Software Instance type</td>
</tr>
<tr>
<td>PatchNumber</td>
<td>Software Instance patch</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Software Instance service_pack</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Virtualization OS&quot;</td>
</tr>
</tbody>
</table>

**BMC_VirtualSystemEnabler relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>VIRTUALSYSTEMOS</td>
<td>BMC_VirtualSystemEnabler</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>HOSTEDSYSTEMCOMPONENTS</td>
<td>BMC_Mainframe / BMC_ComputerSystem (containing)</td>
<td>BMC_VirtualSystemEnabler</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_VirtualSystemEnabler</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_Mainframe / BMC_ComputerSystem (containing)</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
</tbody>
</table>

**BMC_Sysplex / BMC_Cluster**

The root MFPart is related to a Cluster node to represent its sysplex. This is mapped to BMC_Sysplex in CMDB versions prior to 7.6.03, and to BMC_Cluster in version 7.6.03. In either case, the attributes are populated as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Cluster key</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Cluster name</td>
</tr>
<tr>
<td>Description</td>
<td>Cluster description</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>&quot;Sysplex&quot;</td>
</tr>
<tr>
<td>Types</td>
<td>&quot;SysPlex&quot;</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Cluster&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Mainframe&quot;</td>
</tr>
</tbody>
</table>

#### BMC_Sysplex / BMC_Cluster relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>CLUSTEREDSYSTEM</td>
<td>BMC_Sysplex / BMC_Cluster</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_Sysplex</td>
<td>BMC_OperatingSystem</td>
</tr>
</tbody>
</table>

#### BMC_MFCouplingFacility

The MFPart and Cluster are both related to a number of Coupling Facility nodes. Each one is mapped to a BMC_MFCouplingFacility CI as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Coupling Facility key</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>Description</td>
<td>Coupling Facility description</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Coupling Facility name</td>
</tr>
<tr>
<td>CFRMName</td>
<td>Coupling Facility cfrm_name</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>&quot;CF&quot;</td>
</tr>
<tr>
<td>NodeDescriptor</td>
<td>Coupling Facility node_descriptor</td>
</tr>
<tr>
<td>Storage</td>
<td>Coupling Facility storage</td>
</tr>
<tr>
<td>Category</td>
<td>Hardware</td>
</tr>
<tr>
<td>Type</td>
<td>Processing Unit</td>
</tr>
<tr>
<td>Item</td>
<td>Mainframe</td>
</tr>
</tbody>
</table>

#### BMC_MFCouplingFacility relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>COUPLINGFACILITYOS</td>
<td>BMC_MFCouplingFacility</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSPLEXCOUPLINGFACILITY</td>
<td>BMC_Sysplex / BMC_Cluster</td>
<td>BMC_MFCouplingFacility</td>
</tr>
</tbody>
</table>
BMC_StorageSubsystem

The root MFPart can be related to a number of StorageCollection nodes. Each node is mapped to a BMC_StorageSubsystem CI as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>StorageCollection key</th>
</tr>
</thead>
<tbody>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>StorageCollection name</td>
</tr>
<tr>
<td>Description</td>
<td>StorageCollection key</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>StorageCollection vendor</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>StorageCollection type</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>StorageSubsystem</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Disk device&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Disk array&quot;</td>
</tr>
</tbody>
</table>

BMC_StorageSubsystem relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>STORAGESUBSYSTEMOS</td>
<td>BMC_StorageSubsystem</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_StorageSubsystem</td>
<td>BMC_OperatingSystem</td>
</tr>
</tbody>
</table>

BMC_DiskDrive / BMC_TapeDrive

Both the root MFPart and the StorageCollection nodes are related to Storage nodes, representing disk and tape drives. Disk drives are mapped to BMC_DiskDrive CIs as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Storage key</th>
</tr>
</thead>
<tbody>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Storage name</td>
</tr>
<tr>
<td>Description</td>
<td>Storage key</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Storage vendor</td>
</tr>
<tr>
<td>MediaType</td>
<td>1 (fixed hard disk)</td>
</tr>
</tbody>
</table>
Tape drives are mapped to **BMC_TapeDrive** CIs as follows:

<table>
<thead>
<tr>
<th>Name</th>
<th>Storage key</th>
</tr>
</thead>
<tbody>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Storage name</td>
</tr>
<tr>
<td>Description</td>
<td>Storage key</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Storage vendor</td>
</tr>
<tr>
<td>MediaType</td>
<td>0 (removable media)</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>Storage model</td>
</tr>
<tr>
<td>PNPDeviceID</td>
<td>Storage device_id</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Tape device&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Tape drive&quot;</td>
</tr>
</tbody>
</table>

**BMC_DiskDrive / BMC_TapeDrive relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>STORAGESUBSYSTEMDASD</td>
<td>BMC_StorageSubsystem</td>
<td>BMC_DiskDrive</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>DASDOS</td>
<td>BMC_DiskDrive</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>STORAGESUBSYSTEMTAPE</td>
<td>BMC_StorageSubsystem</td>
<td>BMC_TapeDrive</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>TAPEOS</td>
<td>BMC_TapeDrive</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_DiskDrive</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_TapeDrive</td>
<td>BMC_OperatingSystem</td>
</tr>
</tbody>
</table>
BMC_MFSoftwareServer / BMC_SoftwareServer

Each Software Instance related to the root MFPart is mapped to a corresponding CI. In CMDB versions prior to 7.6.03, the CI has the extension class BMC_MFSoftwareServer; in CMDB version 7.6.03, the CI has the base BMC_SoftwareServer class. In both cases, the attributes are as follows:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Instance key</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Software Instance name</td>
</tr>
<tr>
<td>Description</td>
<td>Software Instance key</td>
</tr>
<tr>
<td>BuildNumber</td>
<td>Software Instance build</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>JobName</td>
<td>Software Instance job_id CMDB versions prior to 7.6.03</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Publisher specified in the maintaining Pattern or on the Software Instance — from TKU 2014-Apr-1 onwards</td>
</tr>
<tr>
<td>MFJobName</td>
<td>Software Instance job_id CMDB versions 7.6.03 and later</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id — MainView identifier used for launch in context</td>
</tr>
<tr>
<td>Model</td>
<td>Software Instance short_type CMDB versions prior to 7.6.03</td>
</tr>
<tr>
<td>Model</td>
<td>Software Instance type CMDB versions 7.6.03 and later</td>
</tr>
<tr>
<td>OtherSoftwareServerType</td>
<td>Software Instance type</td>
</tr>
<tr>
<td>PatchNumber</td>
<td>Software Instance patch</td>
</tr>
<tr>
<td>ServicePack</td>
<td>Software Instance service_pack</td>
</tr>
<tr>
<td>ServerID</td>
<td>Software Instance server_id CMDB versions prior to 7.6.03</td>
</tr>
<tr>
<td>MFServerID</td>
<td>Software Instance server_id CMDB versions 7.6.03 and later</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;BMC Discovered&quot;</td>
</tr>
</tbody>
</table>

BMC_MFSoftwareServer / BMC_SoftwareServer relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMCOMPUTER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMDEPENDENCY</td>
<td>BMC_SoftwareServer</td>
<td>BMC_SoftwareServer (dependant)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(depended upon)</td>
<td></td>
</tr>
<tr>
<td>Relationship</td>
<td>Name</td>
<td>Source</td>
<td>Destination</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------</td>
<td>--------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_SoftwareServer</td>
</tr>
</tbody>
</table>

**BMC_ApplicationService**

SoftwareComponent is mapped to BMC_ApplicationService.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Software Component name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Software Component name</td>
</tr>
<tr>
<td>Description</td>
<td>Software Component name</td>
</tr>
<tr>
<td>Model</td>
<td>Software Component type</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Software Component version</td>
</tr>
<tr>
<td>ApplicationServiceType</td>
<td>Enumeration value representing the service type</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application Service&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;BMC Discovered&quot;</td>
</tr>
</tbody>
</table>

**Application service relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_ApplicationSystemServices</td>
<td>APPLICATIONSYSTEMSERVICES</td>
<td>BMC_SoftwareServer</td>
<td>BMC_ApplicationService</td>
</tr>
</tbody>
</table>

**BMC_Application**

BusinessApplicationInstance is mapped to BMC_Application. Contained SoftwareInstance and BusinessApplicationInstance nodes are mapped to corresponding BMC_Component relationships.

<table>
<thead>
<tr>
<th>Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>Description</td>
<td>Business Application Instance name</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Business Application Instance product_version</td>
</tr>
<tr>
<td>Model</td>
<td>Business Application Instance type</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Business Application Instance version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Application Platform&quot;</td>
</tr>
</tbody>
</table>
Application relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>APPLICATIONSYSTEMCOMPUTER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_Application</td>
<td>BMC_SoftwareServer / BMC_MFSoftwareServer</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>APPLICATIONSYSTEMHIERARCHY</td>
<td>BMC_Application</td>
<td>BMC_Application (contained)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_Application</td>
<td>BMC_Application (containing)</td>
</tr>
</tbody>
</table>

BMC_DataBase

DatabaseDetail nodes representing logical databases are mapped to BMC_DataBase:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>DatabaseDetail name</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>DatabaseDetail instance</td>
</tr>
<tr>
<td>Description</td>
<td>DatabaseDetail name</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>&quot;IBM&quot;</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id</td>
</tr>
<tr>
<td>Model</td>
<td>DatabaseDetail type</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>DatabaseDetail instance</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Miscellaneous&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Instance&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Database&quot;</td>
</tr>
</tbody>
</table>

BMC_DataBase relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>MANAGEDDATABASE</td>
<td>BMC_SoftwareServer</td>
<td>BMC_DataBase</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_DataBase</td>
</tr>
</tbody>
</table>

BMC_SystemResource

Resources such as table spaces inside databases are represented as DatabaseDetail nodes. These are mapped to BMC_SystemResource CIs:
MQ resources are represented as Detail nodes related to MQ Software Instance nodes. These are also mapped to BMC_SystemResource OIs:
BMC_Dependency

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>DATABASEINDEXSPACE</td>
<td>BMC_DataBase</td>
<td>BMC_SystemResource</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>DATABASEPAGESET</td>
<td>BMC_DataBase</td>
<td>BMC_SystemResource</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>DATABASETABLESPACE</td>
<td>BMC_DataBase</td>
<td>BMC_SystemResource</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_DataBase</td>
<td>BMC_Transaction</td>
</tr>
</tbody>
</table>

**BMC_Transaction**

Transactions within IMSTM and CICS are represented as Detail nodes related to Software Instance nodes. These are mapped to BMC_Transaction:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Detail key</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;Mainframe&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Detail name</td>
</tr>
<tr>
<td>Description</td>
<td>Detail key</td>
</tr>
<tr>
<td>ComponentAliases</td>
<td>Component alias from mainframe discovery</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>&quot;IBM&quot;</td>
</tr>
<tr>
<td>MFIntegrationID</td>
<td>MFPart mf_integration_id</td>
</tr>
<tr>
<td>Model</td>
<td>Detail type</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>Detail instance</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Unknown&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Unknown&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;BMC Discovered&quot;</td>
</tr>
</tbody>
</table>

**Details nodes not created by default**

Details nodes are not created by default. To enable their creation, you must enable the transaction discovery method described in Enabling the remaining mainframe discovery methods.

**BMC_Transaction relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>SERVERTRANSACTION</td>
<td>BMC_SoftwareServer</td>
<td>BMC_Transaction</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_SoftwareServer</td>
<td>BMC_Transaction</td>
</tr>
</tbody>
</table>
BMC_ConcreteCollection

FunctionalComponent nodes representing CAM Functional Components are mapped to BMC_ConcreteCollection.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Functional Component name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Functional Component name</td>
</tr>
<tr>
<td>Description</td>
<td>Functional Component name</td>
</tr>
<tr>
<td>Model</td>
<td>Functional Component type</td>
</tr>
<tr>
<td>Company</td>
<td>MFPart Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Application&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Functional Component&quot;</td>
</tr>
</tbody>
</table>

Functional Component relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_SoftwareServer</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_DataBase</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_Application</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>ITEMINFUNCTIONALCOMPONENT</td>
<td>BMC_ConcreteCollection</td>
<td>BMC_ApplicationService</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>FUNCTIONALCOMPONENTINAPPLICATION</td>
<td>BMC_Application</td>
<td>BMC_ConcreteCollection</td>
</tr>
</tbody>
</table>

The relationship between BMC_Application and BMC_ConcreteCollection is used for Functional Components reached from any of the following node kinds:

- SoftwareInstance
- DatabaseDetail
- BAI
- SoftwareComponent

CDM Mapping for Network Device

This topic provides information on mapping for network devices.

- BMC_ComputerSystem
- BMC_ComputerSystem (Host Container)
- BMC_HardwareSystemComponent
- BMC_OperatingSystem
- BMC_NetworkPort, BMC_LANEndpoint and BMC_IPEndpoint
- BMC_IPConnectivitySubnet
Network Device nodes in the Discovery model are mapped to the Common Data Model as shown in this automatically-generated diagram:

This diagram illustrates how the Network Device nodes in the Discovery model are mapped to the Common Data Model.

The CIs created are described in this topic.

Load balancers are mapped as Network Devices. The mapping is described in CDM Mapping for Load Balancer nodes.

Data models

### CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name &quot;ImpactOnly&quot; are created.</td>
</tr>
</tbody>
</table>
### Data model and CMDB versions

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>

### BMC_ComputerSystem

The root Network Device node is mapped to a root BMC_ComputerSystem CI with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Network Device name attribute. The configured name of the network device</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Network Device name</td>
</tr>
<tr>
<td>Description</td>
<td>Network Device name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>List of device capabilities</td>
</tr>
<tr>
<td>HostName</td>
<td>Network Device name</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>Network Device last_update_success — the date and time the device was last scanned</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Network Device vendor</td>
</tr>
<tr>
<td>Model</td>
<td>Network Device model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Switch, Router, or other network device enumeration value</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>Network Device serial</td>
</tr>
<tr>
<td>PartitionID</td>
<td>NetworkDevice partition_id</td>
</tr>
<tr>
<td>SystemOID</td>
<td>Network Device sysobjectid</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Switch&quot; or &quot;Router&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Data Switch&quot; or &quot;Access Router&quot;</td>
</tr>
</tbody>
</table>

### Relationships

Relationships are created to represent the connection between edge switches and hosts. Switch-to-switch connections are not mapped.

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>NETWORKLINK</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
</tbody>
</table>
BMC_ComputerSystem (Host Container)

HostContainer nodes directly connected to the root Network Device in the Discovery model are mapped to BMC_ComputerSystem CIs representing the partitioned container.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Host Container name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Host Container name</td>
</tr>
<tr>
<td>Description</td>
<td>Host Container name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>“Blade System” (48) or “Other” (2)</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>Host last_update_success</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Host Container vendor</td>
</tr>
<tr>
<td>Model</td>
<td>Host Container model</td>
</tr>
<tr>
<td>OtherCapabilityDescription</td>
<td>Host Container type</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>“Blade System” (48) or “Other” (2)</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>Host Container serial</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>HadwareContainer type</td>
</tr>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: hashedkey&quot;, where hashedkey is a hash of HostContainer key</td>
</tr>
</tbody>
</table>

Host containment relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>SYSTEMPARTITION</td>
<td>BMC_ComputerSystem (container)</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (container)</td>
<td>BMC_ComputerSystem (contained)</td>
</tr>
</tbody>
</table>

BMC_HardwareSystemComponent

Any ManagementController for a HostContainer containing the root Network Device in the Discovery model is mapped to BMC_HardwareSystemComponent.

⚠️ There is no way to map the ManagementController's IP information to the CDM.

BMC_HardwareSystemComponent

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Management controller name.</td>
</tr>
</tbody>
</table>
Attribute | Details
--- | ---
NameFormat | "HostName"
ShortDescription | Management controller name.
Description | Vendor name, controller model and type. With serial if known.
ManufacturerName | Management controller vendor name.
VersionNumber | Management controller version number.
Model | Management controller model.
PowerManagementSupported | Yes (0) or null if not known.
Company | Host Company.
Category | "Hardware"
Type | "Hardware"
Item | "Management controller" where managed element is a directly connected host. "Chassis manager" where the managed element is a chassis.

Hardware system component relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_HardwareSystemComponent</td>
</tr>
</tbody>
</table>

Corresponds to a Host or a HostContainer, depending on whether the ManagementController controls a chassis, or a host directly.

**BMC_OperatingSystem**

The root Network Device node is also mapped to a **BMC_OperatingSystem** CI:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>&quot;os_vendor os_type os_version&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>&quot;os_type os_version&quot;</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;os_vendor os_type os_version&quot;</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Network Device os_vendor</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>Network Device os_version</td>
</tr>
<tr>
<td>Model</td>
<td>Network Device os_type</td>
</tr>
<tr>
<td>OSType</td>
<td>Other (0)</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>Network Device os_version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
</tbody>
</table>
### BMC_OperatingSystem relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMOS</td>
<td>BMC_ComputerSystem</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_OperatingSystem</td>
<td>BMC_ComputerSystem</td>
</tr>
</tbody>
</table>

### BMC_NetworkPort, BMC_LANEndpoint and BMC_IPEndpoint

Each Network Interface node connected to the root Network Device in the Discovery model is mapped to a BMC_NetworkPort, and its MAC address is mapped to a BMC_LANEndpoint.

Where Discovery has been able to connect a Network Device to a Host, a BMC_Dependency relationship is created between the two BMC_ComputerSystem CIs representing the host and the network device.

### BMC_NetworkPort

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>Description</td>
<td>NetworkInterface name”</td>
</tr>
<tr>
<td>AutoSense</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>FullDuplex</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>NetworkInterface manufacturer</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>PortType</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>Speed</td>
<td>NetworkInterface raw_speed</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Network interface card&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

BMC_NetworkPort was not created for Network Device nodes in versions prior to 9.0.
**BMC_LANEndpoint**

MAC addresses in the Discovery data model are stored in the conventional form with colons separating each pair of hexadecimal digits; in the CDM, MAC addresses are stored as just the hexadecimal digits, with no separating colons.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;MAC&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;MAC address on hostname&quot;</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>Ethernet (14)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;MAC Address&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

**BMC_LANEndpoint** was not created for Network Device nodes in versions prior to 9.0.

**BMC_IPEndpoint**

Each IP Address node related to the root Network Device node is mapped to a **BMC_IPEndpoint** CI:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>Description</td>
<td>IP Address name</td>
</tr>
<tr>
<td>Address</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>DNSHostName</td>
<td>IP Address fqdn (first item in list)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>IPv4 (2) or IPv6 (3)</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ManagementAddress</td>
<td>Yes (1) if the IP address was used to scan the host; No (0) if not. (Not in CMDB 7.5.)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;IP Address&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

**BMC_IPEndpoint** was not created for Network Device nodes in versions prior to 9.0.

### NetworkPort and Endpoint relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_NetworkPort</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>BINDSTO</td>
<td>BMC_LANEndpoint</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>NETWORKLINK</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
</tbody>
</table>

### BMC_IPConnectivitySubnet

**Discovery** Subnet is mapped to **BMC_IPConnectivitySubnet**:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Description</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IP Address prefix_length</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>SubnetNumber</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>&quot;Subnet&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;TCP/IP&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

BMC_IPConnectivitySubnet was not created for Network Device in versions prior to 9.0.

### Subnet relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_InIPSubnet</td>
<td>INIPSUBNET</td>
<td>BMC_IPConnectivitySubnet</td>
<td>BMC_IPEndPoint</td>
</tr>
</tbody>
</table>

### CDM Mapping for Load Balancer nodes

LoadBalancerInstance nodes are mapped to details on the base Network Device. The mappings for the remaining load balancer components are described in this topic.

#### Load balancer nodes

- LoadBalancerInstance nodes are mapped to details on the base Network Device.
- LoadBalancerService nodes are mapped to BMC_LogicalSystemComponent CIs.
- LoadBalancerPool nodes are mapped to BMC_ResourcePool CIs.

### Data models

#### CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.
Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name “ImpactOnly” are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>

**BMC_LogicalSystemComponent**

LoadBalancerService nodes are mapped to BMC_LogicalSystemComponent CIs.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: hashedkey&quot;, where hashedkey is a hash of the LoadBalancerService key.</td>
</tr>
<tr>
<td>Name</td>
<td>Load Balancer Service Name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Load Balancer Service Name</td>
</tr>
<tr>
<td>Description</td>
<td>Load Balancer Service Name</td>
</tr>
<tr>
<td>Company</td>
<td>Load Balancer Service Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Service&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Load balancer service&quot;</td>
</tr>
</tbody>
</table>

**BMC_LogicalSystemComponent relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERSERVICE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_LogicalSystemComponent</td>
<td>Source</td>
</tr>
</tbody>
</table>

**BMC_ResourcePool**

LoadBalancerPool nodes are mapped to BMC_ResourcePool CIs.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: hashedkey&quot;, where hashedkey is a hash of the LoadBalancerPool key.</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Load Balancer Pool Name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Load Balancer Pool Name</td>
</tr>
<tr>
<td>Description</td>
<td>Load Balancer Pool Name</td>
</tr>
<tr>
<td>ResourceType</td>
<td>Other (0)</td>
</tr>
<tr>
<td>OtherResourceType</td>
<td>&quot;Load Balancer Pool&quot;</td>
</tr>
<tr>
<td>Company</td>
<td>Load Balancer Pool Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Service&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Load balancer pool&quot;</td>
</tr>
</tbody>
</table>

### Host to BMC_ResourcePool Relationship

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>HOSTEDRESOURCEPOOL</td>
<td>BMC_ComputerSystem (Host)</td>
<td>BMC_ResourcePool</td>
</tr>
</tbody>
</table>

### BMC_ResourcePool Relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERSERVICEPOOL</td>
<td>BMC_ResourcePool</td>
<td>BMC_LogicalSystemComponent</td>
<td>Source</td>
</tr>
</tbody>
</table>

### BMC_Cluster

Load Balancer Group nodes are mapped as BMC_Cluster CIs.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: hashedkey&quot;, where hashedkey is a hash of the LoadBalancerGroup key.</td>
</tr>
<tr>
<td>Name</td>
<td>Load Balancer Group name</td>
</tr>
<tr>
<td>Description</td>
<td>Load Balancer Group name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Load Balancer Group name</td>
</tr>
<tr>
<td>Model</td>
<td>Load Balancer Group type</td>
</tr>
<tr>
<td>Company</td>
<td>computersystem.Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Load balancer cluster&quot;</td>
</tr>
</tbody>
</table>

### BMC_Cluster relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>CLUSTEREDSYSTEM</td>
<td>BMC_Cluster</td>
<td></td>
<td>Destination</td>
</tr>
<tr>
<td>Relationship</td>
<td>Name</td>
<td>Source</td>
<td>Destination</td>
<td>Impact from</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>-----------------------</td>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>BMC_ComputerSystem</td>
<td>(Network Device)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**BMC_ResourcePool linking**

BMC_ResourcePool CIs are linked to BMC_ApplicationService, BMC_SoftwareServer, BMC_ApplicationSystem or BMC_ComputerSystem CIs.

**BMC_ApplicationService**

BMC_ApplicationService CIs (mapped from Software Components) are linked to BMC_ResourcePool (mapped from Load Balancer Pool nodes).

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERMEMBER</td>
<td>BMC_ApplicationService</td>
<td>BMC_ResourcePool</td>
<td>Source</td>
</tr>
</tbody>
</table>

**BMC_SoftwareServer and BMC_ApplicationSystem**

BMC_SoftwareServer CIs (mapped from First-order Software Instances) are linked to BMC_ResourcePool (mapped from Load Balancer Pool nodes).

BMC_ApplicationSystem CIs (mapped from Second-order Software Instances) are linked to BMC_ResourcePool (mapped from Load Balancer Pool nodes).

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERMEMBER</td>
<td>BMC_SoftwareServer</td>
<td>BMC_ResourcePool</td>
<td>Source</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERMEMBER</td>
<td>BMC_ApplicationSystem</td>
<td>BMC_ResourcePool</td>
<td>Source</td>
</tr>
</tbody>
</table>

**BMC_ComputerSystem**

BMC_ComputerSystem CIs (mapped from Hosts) are linked to BMC_ResourcePool (mapped from Load Balancer Pool nodes) only if there is no software link.

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>LOADBALANCERMEMBER</td>
<td>BMC_ComputerSystem</td>
<td>BMC_ResourcePool</td>
<td>Source</td>
</tr>
</tbody>
</table>

**CDM Mapping for Printer**

This topic provides information about mapping for printer nodes.

- **BMC_Printer**
- **BMC_NetworkPort, BMC_LANEndpoint and BMC_IPEndPoint**
- **BMC_IPConnectivitySubnet**

Printer nodes in the Discovery model are mapped to the Common Data Model as shown in this automatically-generated diagram:
This diagram illustrates how the Printer nodes in the Discovery model are mapped to the Common Data Model.

The CIs created are described in this topic.

Data models

### CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name “ImpactOnly” are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>

**BMC_Printer**

The root Printer node is mapped to a root BMC_Printer CI with the following attributes:
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Printer name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Printer name</td>
</tr>
<tr>
<td>Description</td>
<td>Printer name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>Print</td>
</tr>
<tr>
<td>HostName</td>
<td>Printer sysname</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>Printer last_update_success — the date and time the device was last scanned</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Printer vendor</td>
</tr>
<tr>
<td>Model</td>
<td>Printer model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Print (11)</td>
</tr>
<tr>
<td>PrinterType</td>
<td>Network Printer (3)</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>Printer serial</td>
</tr>
<tr>
<td>SystemOID</td>
<td>Printer sysobjectid</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Peripheral&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Printer&quot;</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort, BMC_LANEndpoint and BMC_IPEndpoint**

Each Network Interface node connected to the root Printer node in the Discovery model is mapped to a **BMC_NetworkPort**, and its MAC address is mapped to a **BMC_LANEndpoint**.

Where Discovery has been able to connect a Printer node to a Network Device node, a **BMC_Dependency** relationship is created between the two **BMC_ComputerSystem** CIs representing the printer and the network device.

### BMC_NetworkPort

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Network Interface interface_name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Network Interface interface_name</td>
</tr>
<tr>
<td>Description</td>
<td>Network Interface name&quot;</td>
</tr>
<tr>
<td>AutoSense</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>FullDuplex</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Network Interface manufacturer</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>Network Interface mac_addr</td>
</tr>
</tbody>
</table>
**Attribute Details**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>PortType</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>Speed</td>
<td>Network Interface raw_speed</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Network interface card&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

**BMC_NetworkPort** was not created for Printer nodes in versions prior to 9.0.

**BMC_LANEndpoint**

MAC addresses in the Discovery data model are stored in the conventional form with colons separating each pair of hexadecimal digits; in the CDM, MAC addresses are stored as just the hexadecimal digits, with no separating colons.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Network Interface mac_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;MAC&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Network Interface mac_addr</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;MAC address on hostname&quot;</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>Ethernet (14)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;MAC Address&quot;</td>
</tr>
</tbody>
</table>

⚠️ **Differences from earlier BMC Atrium Discovery versions**

**BMC_LANEndpoint** was not created for Printer in versions prior to 9.0.

**BMC_IPEndpoint**

Each IP Address node related to the root Printer node is mapped to a **BMC_IPEndpoint CI**.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>IP Address <code>ip_addr</code></td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>IP Address <code>ip_addr</code></td>
</tr>
<tr>
<td>Description</td>
<td>IP Address <code>name</code></td>
</tr>
<tr>
<td>Address</td>
<td>IP Address <code>ip_addr</code></td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>DNSHostName</td>
<td>IP Address <code>fqdn</code> (first item in list)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>IPv4 (2) or IPv6 (3)</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address <code>netmask</code></td>
</tr>
<tr>
<td>ManagementAddress</td>
<td>Yes (1) if the IP address was used to scan the host; No (0) if not. (Not in CMDB 7.5.)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;IP Address&quot;</td>
</tr>
</tbody>
</table>

⚠️ Differences from earlier BMC Atrium Discovery versions

`BMC_IPEndpoint` was not created for Printer nodes in versions prior to 9.0.

NetworkPort and Endpoint relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_NetworkPort</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>BINDSTO</td>
<td>BMC_LANEndpoint</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>NETWORKLINK</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_ComputerSystem (network device)</td>
<td>BMC_ComputerSystem (host)</td>
</tr>
</tbody>
</table>

**BMC_IPConnectivitySubnet**

Discovery Subnet nodes are mapped to `BMC_IPConnectivitySubnet`: 
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Description</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IP Address prefix_length</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>SubnetNumber</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Subnet&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;TCP/IP&quot;</td>
</tr>
</tbody>
</table>

⚠️ Differences from earlier BMC Atrium Discovery versions

BMC_IPConnectivitySubnet was not created for Printer nodes in versions prior to 9.0.

Subnet relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_InIPSubnet</td>
<td>INIPSUBNET</td>
<td>BMC_IPConnectivitySubnet</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>

CDM Mapping for SNMP Managed Device

This topic provides information about mapping for SNMP managed devices.

- BMC_ComputerSystem
- BMC_OperatingSystem
- BMC_NetworkPort, BMC_IPEndpoint and BMC_LANEndpoint
- BMC_IPConnectivitySubnet

SNMPManagedDevice nodes in the Discovery model are mapped to the Common Data Model as shown in this automatically-generated diagram:
This diagram illustrates how the SNMPManagedDevice nodes in the Discovery model are mapped to the Common Data Model.

The CIs created are described in this topic.

Data models

## CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name “ImpactOnly” are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>
**BMC_ComputerSystem**

The root SNMPManagedDevice node is mapped to a root BMC_ComputerSystem CI with the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>SNMPManagedDevice name attribute. The configured name of the SNMP managed device</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>SNMPManagedDevice name</td>
</tr>
<tr>
<td>Description</td>
<td>SNMPManagedDevice name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>List of device capabilities</td>
</tr>
<tr>
<td>HostName</td>
<td>SNMPManagedDevice name</td>
</tr>
<tr>
<td>LastScanDate</td>
<td>SNMPManagedDevice last_update_success — the date and time the device was last scanned</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>SNMPManagedDevice vendor</td>
</tr>
<tr>
<td>Model</td>
<td>SNMPManagedDevice model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Switch, Router, or other network device enumeration value</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>SNMPManagedDevice serial</td>
</tr>
<tr>
<td>System OID</td>
<td>SNMPManagedDevice sysobjectid</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Switch&quot; or &quot;Router&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Data Switch&quot; or &quot;Access Router&quot;</td>
</tr>
</tbody>
</table>

**BMC_OperatingSystem**

The root SNMPManagedDevice node is also mapped to a BMC_OperatingSystem CI:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>&quot;os_vendor os_type os_version&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>&quot;os_type os_version&quot;</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;os_vendor os_type os_version&quot;</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>SNMPManagedDevice os_vendor</td>
</tr>
<tr>
<td>MarketVersion</td>
<td>SNMPManagedDevice os_version</td>
</tr>
<tr>
<td>Model</td>
<td>SNMPManagedDevice os_type</td>
</tr>
<tr>
<td>OSTYPE</td>
<td>Other (0)</td>
</tr>
<tr>
<td>VersionNumber</td>
<td>SNMPManagedDevice os_version</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Software&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Operating System Software&quot;</td>
</tr>
</tbody>
</table>
**BMC_OperatingSystem relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMOS</td>
<td>BMC_ComputerSystem</td>
<td>BMC_OperatingSystem</td>
</tr>
<tr>
<td>BMC_Impact</td>
<td>IMPACT</td>
<td>BMC_OperatingSystem</td>
<td>BMC_ComputerSystem</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort, BMC_IPEndpoint and BMC_LANEndpoint**

Each NetworkInterface node connected to the root SNMPManagedDevice in the Discovery model is mapped to a BMC_NetworkPort, and its MAC address is mapped to a BMC_LANEndpoint.

**BMC_NetworkPort**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface interface_name</td>
</tr>
<tr>
<td>Description</td>
<td>NetworkInterface name</td>
</tr>
<tr>
<td>AutoSense</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>FullDuplex</td>
<td>Yes (0) or No (1) or null if not known</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>NetworkInterface manufacturer</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>PortType</td>
<td>Ethernet (2)</td>
</tr>
<tr>
<td>Speed</td>
<td>NetworkInterface raw_speed</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Network interface card&quot;</td>
</tr>
</tbody>
</table>

**BMC_LANEndpoint**

MAC addresses in the Discovery data model are stored in the conventional form with colons separating each pair of hexadecimal digits; in the CDM, MAC addresses are stored as just the hexadecimal digits, with no separating colons.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;MAC&quot;</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShortDescription</td>
<td>NetworkInterface mac_addr</td>
</tr>
<tr>
<td>Description</td>
<td>&quot;MAC address on hostname&quot;</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>Ethernet (14)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;MAC Address&quot;</td>
</tr>
</tbody>
</table>

**BMC_IPEndpoint**

Each IPAddress node related to the root SNMP Managed Device node is mapped to a BMC_IPEndpoint CI:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>Description</td>
<td>IP Address name</td>
</tr>
<tr>
<td>Address</td>
<td>IP Address ip_addr</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>DNSHostName</td>
<td>IP Address fqdns (first Item in list)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>IPv4 (2) or IPv6 (3)</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>ManagementAddress</td>
<td>Yes (1) if the IP address was used to scan the host; No (0) if not. (Not in CMDB 7.5.)</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;IP Address&quot;</td>
</tr>
</tbody>
</table>

**NetworkPort and Endpoint relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_NetworkPort</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>
### BMC_IPConnectivitySubnet

**Discovery** Subnet is mapped to **BMC_IPConnectivitySubnet**:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Description</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IP Address prefix_length</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IP Address netmask</td>
</tr>
<tr>
<td>SubnetNumber</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Subnet&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;TCP/IP&quot;</td>
</tr>
</tbody>
</table>

**Subnet relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_InIPSubnet</td>
<td>INIPSUBNET</td>
<td>BMC_IPConnectivitySubnet</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>

### CDM Mapping for Storage

This topic provides information about mapping for storage.

- **BMC_ComputerSystem**
- **BMC_ResourcePool**
- **BMC_HardwarePackage**
- **BMC_NetworkPort, BMC_IPEndpoint and BMC_LANEndpoint**
- **BMC_IPConnectivitySubnet**
- **BMC_NetworkPort**
- **BMC_Cluster**
- **Storage Systems links to consuming hosts**

The CIs created are described in this topic.
Data models

CMDB version differences

The default mapping works with CMDB versions 7.5, 7.6, and 7.6.03 and later. The data model in each is slightly different.

- BMC_Impact relationships are not normally created with CMDB 7.6.03 and later, since the CMDB automatically maintains them itself. See the information about impact relationships for more details.
- A number of attributes are not present in CMDB 7.5, as noted below.

Click here for reference information on impact and the data models.

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name &quot;ImpactOnly&quot; are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name &quot;IMPACT&quot; are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name &quot;IMPACT&quot; are created.</td>
</tr>
</tbody>
</table>

BMC_ComputerSystem

The StorageSystem node is mapped to the BMC_ComputerSystem CI.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>StorageSystem name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>StorageSystem name</td>
</tr>
<tr>
<td>Description</td>
<td>StorageSystem name</td>
</tr>
<tr>
<td>CapabilityList</td>
<td>List of capabilities</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>StorageSystem vendor</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>StorageSystem type:StorageSystem model</td>
</tr>
<tr>
<td>PrimaryCapability</td>
<td>Storage</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>StorageSystem serial number</td>
</tr>
<tr>
<td>Category</td>
<td>The CDM category, for example, &quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>The CDM type, for example, &quot;Disk device&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>The CDM item, for example, &quot;Disk array&quot;</td>
</tr>
</tbody>
</table>

### BMC_ComputerSystem relationships

Populate the Company attribute for BMC_ComputerSystem CIs mapped from StorageSystem nodes.

Creates links to manager CI and sets LastScanDate and SystemOID where available.

![Important note]

Do not create an extension syncmapping that extends this one. If you need to access nodes accessed in the traversals here, traverse again. This priority syncmapping must complete before any other CIs are created.

- First-order Software Instances are mapped to BMC_SoftwareServer CIs.
- Second-order Software Instances are mapped to BMC_ApplicationSystem CIs.

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>SYSTEMMANAGER</td>
<td>BMC_SoftwareServer</td>
<td>BMC_ComputerSystem</td>
<td>Source</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>SYSTEMMANAGER</td>
<td>BMC_ApplicationSystem</td>
<td>BMC_ComputerSystem</td>
<td>Source</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>SYSTEMMANAGER</td>
<td>BMC_SoftwareServer</td>
<td>BMC_ComputerSystem</td>
<td></td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>SYSTEMMANAGER</td>
<td>BMC_ApplicationSystem</td>
<td>BMC_ComputerSystem</td>
<td></td>
</tr>
</tbody>
</table>

### BMC_ResourcePool

The StoragePool node is mapped to the BMC_ResourcePool CI.

### BMC_ResourcePool

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM:StoragePool key&quot;.</td>
</tr>
<tr>
<td>Name</td>
<td>StoragePool name</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>StoragePool ID</td>
</tr>
<tr>
<td>Description</td>
<td>StoragePool name</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>AllocationUnits</td>
<td>&quot;byte&quot;</td>
</tr>
<tr>
<td>Capacity</td>
<td>StoragePool capacity</td>
</tr>
<tr>
<td>CurrentlyConsumedResource</td>
<td>StoragePool consumed capacity</td>
</tr>
<tr>
<td>Primordial</td>
<td>If the pool type is primordial, &quot;Primordial&quot;</td>
</tr>
<tr>
<td>ResourceType</td>
<td>32=Storage Volume</td>
</tr>
<tr>
<td>Company</td>
<td>StorageSystem Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Storage&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Storage pool&quot;</td>
</tr>
</tbody>
</table>

**BMC_ResourcePool relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>HOSTEDRESOURCEPOOL</td>
<td>BMC_ComputerSystem</td>
<td>BMC_ResourcePool</td>
</tr>
</tbody>
</table>

**BMC_HardwarePackage**

The StorageProcessor node is mapped to the BMC_HardwarePackage CI.

⚠️ From June TKU 2015 onwards Serial number attribute is added for StorageProcessor.

**BMC_HardwarePackage**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: StorageProcessor key&quot;.</td>
</tr>
<tr>
<td>Name</td>
<td>StorageProcessor description.</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>StorageProcessor identifier.</td>
</tr>
<tr>
<td>Description</td>
<td>StorageProcessor description.</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>StorageProcessor vendor.</td>
</tr>
<tr>
<td>Model</td>
<td>StorageProcessor model name.</td>
</tr>
<tr>
<td>Company</td>
<td>StorageProcessor company.</td>
</tr>
<tr>
<td>Serial</td>
<td>StorageProcessor serial number.</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Storage&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Storage processor&quot;</td>
</tr>
</tbody>
</table>
**BMC_HardwarePackage relationships**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Source</th>
<th>Name</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>BMC_ComputerSystem (Storage System)</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_HardwarePackage</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort, BMC_IPEndpoint and BMC_LANEndpoint**

Each Network Interface node connected to the root StorageSystem in the Discovery model is mapped to a BMC_NetworkPort CI, and its MAC address is mapped to a BMC_LANEndpoint CI. Each of its associated IP addresses are mapped to BMC_IPEndpoint CIs.

**BMC_NetworkPort**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Interface name of the Network Interface</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Interface name of the Network Interface</td>
</tr>
<tr>
<td>Description</td>
<td>Full name of the Network Interface</td>
</tr>
<tr>
<td>AutoSense</td>
<td>Negotiation setting, 0=Yes, 1=No.</td>
</tr>
<tr>
<td>FullDuplex</td>
<td>Full duplex, 0=Yes, 1=No.</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>2. (Ethernet)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>The Network Interface manufacturer</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>MAC address.</td>
</tr>
<tr>
<td>PortType</td>
<td>2. (Ethernet)</td>
</tr>
<tr>
<td>Speed</td>
<td>The interface speed in bps</td>
</tr>
<tr>
<td>Company</td>
<td>Network Interface Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Network interface card&quot;</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort relationship**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Source</th>
<th>Name</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedSystemComponents</td>
<td>BMC_ComputerSystem (Storage System)</td>
<td>SYSTEMHARDWARE</td>
<td>BMC_NetworkPort</td>
</tr>
</tbody>
</table>

**BMC_LANEndpoint**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>MAC Address</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;MAC&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>MAC Address (with colons)</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>&quot;MAC address on hostname&quot;</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>MACAddress</td>
<td>MAC address (no separating colons)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>14. (Ethernet)</td>
</tr>
<tr>
<td>Company</td>
<td>Network Interface Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;MAC Address&quot;</td>
</tr>
</tbody>
</table>

#### BMC_LANEndpoint relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem (Storage System)</td>
<td>BMC_LANEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_LANEndpoint</td>
</tr>
</tbody>
</table>

#### BMC_IPEndpoint

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>IP address ip_addr</td>
</tr>
<tr>
<td>NameFormat</td>
<td>&quot;IP&quot;</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>IP address ip_addr</td>
</tr>
<tr>
<td>Description</td>
<td>IP address name</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>Address</td>
<td>IP address ip_addr</td>
</tr>
<tr>
<td>DNSHostName</td>
<td>IP address fqdns (first item in list)</td>
</tr>
<tr>
<td>ManagementAddress</td>
<td>Yes (1) if the IP address was used to scan the host; No (0) if not. (Not in CMDB 7.5.)</td>
</tr>
<tr>
<td>ProtocolType</td>
<td>IPv4 (2) or IPv6 (3)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IPv6 prefix length in bits</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IPv4 subnet mask</td>
</tr>
<tr>
<td>Company</td>
<td>Network Interface Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Address&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;IP Address&quot;</td>
</tr>
</tbody>
</table>

#### BMC_IPEndpoint relationships
### Relationship Table

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_HostedAccessPoint</td>
<td>HOSTEDACCESSPOINT</td>
<td>BMC_ComputerSystem (Storage System)</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>ENDPOINTPORT</td>
<td>BMC_NetworkPort</td>
<td>BMC_IPEndpoint</td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>BINDSTO</td>
<td>BMC_LANEndpoint</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>

### BMC_IPConnectivitySubnet

The Subnet node is mapped to BMC_IPConnectivitySubnet.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>Description</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>SubnetNumber</td>
<td>Subnet ip_address_range</td>
</tr>
<tr>
<td>AddressType</td>
<td>IPv4 (1) or IPv6 (2)</td>
</tr>
<tr>
<td>PrefixLength</td>
<td>IPv6 prefix length in bits</td>
</tr>
<tr>
<td>SubnetMask</td>
<td>IPv4 subnet mask</td>
</tr>
<tr>
<td>Company</td>
<td>Subnet Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Network&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Subnet&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;TCP/IP&quot;</td>
</tr>
</tbody>
</table>

### BMC_IPConnectivitySubnet relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_InIPSubnet</td>
<td>INIPSUBNET</td>
<td>BMC_IPConnectivitySubnet</td>
<td>BMC_IPEndpoint</td>
</tr>
</tbody>
</table>

### BMC_NetworkPort

The front-end FibreChannelPort is mapped to BMC_NetworkPort.

ℹ️ Note that "Fibre Channel" is the correct spelling for the technology. Do not "correct" the item to "Fiber Channel".

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;World Wide Name:World Wide Port Number&quot;.</td>
</tr>
<tr>
<td>Name</td>
<td>Fibre Channel Port name : WWPN</td>
</tr>
<tr>
<td>Attribute</td>
<td>Details</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>Fibre Channel Port name</td>
</tr>
<tr>
<td>Description</td>
<td>Fibre Channel Port name : WWPN</td>
</tr>
<tr>
<td>LinkTechnology</td>
<td>4. (Fibre Channel)</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>Storage processor vendor</td>
</tr>
<tr>
<td>Model</td>
<td>Storage processor model</td>
</tr>
<tr>
<td>PermanentAddress</td>
<td>Fibre Channel Port WWPN</td>
</tr>
<tr>
<td>PortType</td>
<td>6. (Fibre Channel)</td>
</tr>
<tr>
<td>Company</td>
<td>Host Company</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Card&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Fibre Channel HBA card&quot;</td>
</tr>
</tbody>
</table>

**BMC_NetworkPort relationship**

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>SUBHWCOMPONENT</td>
<td>BMC_HardwarePackage</td>
<td>BMC_NetworkPort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Storage Processor)</td>
<td></td>
</tr>
</tbody>
</table>

**BMC_Cluster**

The StorageSystemGroup node is mapped to the BMC_Cluster CI.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenId</td>
<td>Attribute that aids reconciliation of CIs populated by multiple data sources. TokenId is of the form &quot;ADDM: StorageSystemGroup key&quot;.</td>
</tr>
<tr>
<td>Name</td>
<td>StorageSystemGroup name.</td>
</tr>
<tr>
<td>ShortDescription</td>
<td>StorageSystemGroup name.</td>
</tr>
<tr>
<td>Description</td>
<td>StorageSystemGroup name.</td>
</tr>
<tr>
<td>ManufacturerName</td>
<td>StorageSystem vendor.</td>
</tr>
<tr>
<td>Model</td>
<td>StorageSystemGroup type.</td>
</tr>
<tr>
<td>Company</td>
<td>StorageSystemGroup Company.</td>
</tr>
<tr>
<td>Category</td>
<td>&quot;Hardware&quot;</td>
</tr>
<tr>
<td>Type</td>
<td>&quot;Storage&quot;</td>
</tr>
<tr>
<td>Item</td>
<td>&quot;Storage cluster&quot;</td>
</tr>
</tbody>
</table>
### BMC_Cluster relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Component</td>
<td>CLUSTEREDSYSTEM</td>
<td>BMC_Cluster</td>
<td>BMC_ComputerSystem</td>
<td>Destination</td>
</tr>
<tr>
<td>BMC_Component</td>
<td>CLUSTEREDSYSTEM</td>
<td>BMC_Cluster</td>
<td>BMC_ComputerSystem</td>
<td>-</td>
</tr>
</tbody>
</table>

### Storage Systems links to consuming hosts

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Name</th>
<th>Source</th>
<th>Destination</th>
<th>Impact from</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC_Dependency</td>
<td>STORAGEUSE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_ComputerSystem</td>
<td>Source</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Storage System</td>
<td>Consuming Host</td>
<td></td>
</tr>
<tr>
<td>BMC_Dependency</td>
<td>STORAGEUSE</td>
<td>BMC_ComputerSystem</td>
<td>BMC_ComputerSystem</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Storage System</td>
<td>Consuming Host</td>
<td></td>
</tr>
</tbody>
</table>

### Configuring CMDB synchronization through a private queue

Depending on what other integration components (for example, Atrium Integrator, Reconciliation, Normalization) are running on the BMC Remedy AR server to which BMC Atrium Discovery is syncing, there may be a performance benefit to those applications and the sync process from setting up a private queue configuration.

By default BMC Atrium Discovery uses the BMC Atrium CMDB API RPC Queue (390696), which is also the default queue for other integration components. By configuring BMC Atrium Discovery to use one of the private Atrium queues (currently 390698 and 390699 - which are also used for Reconciliation and Normalization CMDB API calls respectively), it may be possible to improve the performance of other components.

The "private" queue is actually a queue shared with Reconciliation or Normalization, so it is especially important to avoid running manual or scheduled Reconciliation and Normalization jobs at the same time as BMC Atrium Discovery synchronizing. Failure to do so could result in these operations completing in indeterminate order, for example data being reconciled to the Asset dataset before being normalized. When populating datasets for the first time, the recommended workflow is to manually run Normalization and Reconciliation jobs following an initial sync, then configure inline normalization as well as continuous reconciliation from that point on (see the [BMC Atrium Core documentation](#) for further details). The best queue to share depends on the amount and timing of activity for each of these three components on the system being configured.

Use the following procedures to manage private queues for CMDB synchronization:

- Configuring a private queue
  - Configure BMC Atrium Discovery to use a private queue
Configuring a private queue

To start CMDB synchronization using a private queue, complete the following steps:

- Configure the use of a private queue on the BMC Remedy AR Server
- Configure BMC Atrium Discovery to use a private queue

⚠️ Warning

Ensure that you avoid a hybrid configuration, where private queues are enabled on the BMC Atrium Discovery appliance but not on the AR server, and the reverse.

To configure the use of private queues on the BMC Remedy AR Server

1. Ensure the following ports appear on the Server Information > Ports and Queues page:
   - 390698
   - 390681
   - 390699
2. From the Server Configuration Editor dialog in the Reconciliation application, set the RPC Socket attribute to 390698.
3. From the System Configuration tab in the Normalization Configuration Editor, set the following attributes as in the following example:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR RPC Queue</td>
<td>390681</td>
</tr>
<tr>
<td>CMDB RPC Queue</td>
<td>390699</td>
</tr>
</tbody>
</table>

4. Restart the BMC Remedy AR Server

Configure BMC Atrium Discovery to use a private queue

⚠️ Note

The configuration given below enables BMC Atrium Discovery to share the Normalization queue. Providing a value of 390698 shares the Reconciliation queue instead.

1. Use the following command to get the ID of the connection for which a private queue is to be configured:

```
$ tw_sync_control -p <system user password> --status
Sync targets status at 2015-06-02 15:44:33
```
2. Configure the queue as follows:

```
$ tw_sync_control -p <system user password> --update --
id=cb3d6b55ea05d193405513776e434d444253796e63436f6e666967 --rpc-queue=390699
Target successfully updated:
  Id: cb3d6b55ea05d193405513776e434d444253796e63436f6e666967
  Name: 8102_2
  Type: CMDB
  User: Demo
  Dataset: Harness
  Datamodel: CMDB 7.6.03 and later (with impact attributes)
  RPC Queue: 390699
Continuous: False
```

3. Use the following command to restart the necessary service:

```
$ sudo service tideway restart cmdb_sync_exporter
```

### Configuring BMC Atrium Discovery to use the default non-private queue

1. Use the following command to reset the CMDB synchronization RPC queue to use the default non-private value (390696):

```
$ tw_sync_control -p <system user password> --update --
id=cb3d6b55ea05d193405513776e434d444253796e63436f6e666967 --rpc-queue=0
```

2. Use the following command to restart the necessary service:

```
$ sudo service tideway restart cmdb_sync_exporter
```
Recreating legacy filters

After upgrade, where filter definitions have not been upgraded automatically, the CMDB Sync page shows a message stating that you must review the filters, and link to the CMDB Sync filtering page where you can do so.

This is also shown on the CMDB Sync filter page, along with a further explanatory message and a link to a summary page which shows the legacy filters for review. The message also provides a button which enables you to delete the old configuration once you have created the new filter definitions.

To recreate legacy filters

From the CMDB Sync filtering page, click the review link in the legacy filter definition warning panel. The Component Filter Summary is displayed. The Component Filter Summary lists the filters that have not been upgraded, grouped by root node type. For each one, the filter condition is listed. In the example below, a single SNMPManagedDevice filter was not upgraded. The condition for this filter is **dns_hostname has subword 'tideway'**

Recreate this filter in the CMDB Sync filter page. To do this:

1. From the CMDB Sync filtering page, select the node kind to filter from the drop-down. In this example, select **SNMPManaged Device**
2. Select the Only SNMPManagedDevice nodes which match the following conditions are synced to the CMDB radio button.
3. Enter the condition from the report in the Query Constructor. In this example the DNS Domain is an attribute on a Network Interface.
   a. In the search box of the first column, enter network. Select Network Interface: Network interfaces of this device from the matches displayed.
   b. In the search box of the first column, enter dns. elect DNS Hostname from the matches displayed.
   c. Select contains word from the drop-down in the DNS hostname condition.
   d. Enter tideway in the DNS hostname condition field.

4. Click Save to save the condition.
5. Repeat the process for each filter listed in the Component Filter Summary.

CMDB Sync Preview

When the CMDB synchronization is configured, you can preview the synchronization results for a set of items using CMDB Sync Preview action. This action is available for hosts, network devices, printers, and some other nodes. You can model the graph of the CIs and relationships that will be populated into the CMDB after synchronization.

This topic contains the following information about viewing and managing the CMDB Sync Preview view:

- Launching the CMDB Sync Preview
- Applying CMDB Filtering
- Viewing key CMDB Sync Preview information
- Minimizing and maximizing software context view
- Zooming and moving items around
- Filtering the software context
- Showing labels
- Changing the CMDB Sync Preview layout
Launching the CMDB Sync Preview

- The CMDB Sync Preview is only supported in modern standards-compliant web browsers. It works best in recent versions of Chrome and Firefox, and it is also supported in Internet Explorer version 10 and later. It is not supported in Internet Explorer 9 or earlier.

You can open CMDB Sync Preview for a custom set of nodes by selecting them in the object list and then using **CMDB Sync Preview** action. By default, the CMDB Sync Preview is maximized.

The following example shows the CMDB Sync Preview of two hosts that were selected manually:
Applying CMDB Filtering

As a part of CMDB Synchronisation configuration for any CMDB connection, the data might be filtered to trim the unwanted elements that should not be synchronized to BMC Atrium CMDB.

As there might be more than one CMDB connection, and filters might vary, CMDB Sync Preview shows data with no filters applied by default.

To see CMDB Sync Preview for the particular CMDB connection, select the necessary CMDB connection in the Apply CMDB Filtering list and click **Refresh Filtering**. The preview adjusts to mark the filtered data.
Explicitly/implicitly filtered objects

Some CIs might be implicitly filtered if they do not match a filtering condition directly, but have a relationship with a CI that does.

Implicitly and implicitly filtered objects are highlighted on the graph:

- Explicitly filtered
- Implicitly filtered

Viewing key CMDB Sync Preview information

To view the key that explains colors and shapes of the nodes and relationships, as well as navigation basics, click following icon:

![Question Mark Icon]

Below is an example of the information that is provided:
Minimizing and maximizing software context view

You can toggle the CMDB Sync Preview window size by switching to a floating pop-up dialog (-) or to a maximized docked panel at the top of the Object Details page (+).

Example of the maximized CMDB Sync Preview for a host node is displayed below:
Zooming and moving items around

In both docked (maximized) and floating (minimized) views, use the mouse scroll wheel to zoom in and out, and drag on the background to move around. On a touch device, pinch to zoom. Double-click inside the view to recenter and zoom the display to fit the window. Drag the background to move the whole diagram. Drag individual icons to rearrange the view.

Filtering the software context

In the maximized view, you can filter the information shown on the CMDB Sync Preview by showing /hiding nodes of some types.

Click on the Show/Hide list, select the node types that you would like to be displayed and deselect those that should be hidden.
Showing labels

The labels are visible only in the maximized view. Tick **Show labels** to show the text label for every node. When labels are hidden, the view is more compact and relationships directions might look clearer. You can see the label hint when you hover over the node.

Changing the CMDB Sync Preview layout

In the maximized view, you can auto-arrange the nodes and relationships layout using the following templates:

- **Switch wide/tall**—Optimizes the graph for horizontal/vertical view.
- **Force-directed**—Arranges nodes so that the impacting nodes have more central position than the impacted ones.
- **Impact**—Levels the impacted objects in several layers depending on the impact depth.

You can also drag nodes manually to tune the layout for the best fit for your specific case.

Enabling BMC Atrium Orchestrator to run commands

This topic provides instructions on enabling BMC Atrium Orchestrator to run commands on a BMC Atrium Discovery appliance. You can do this by installing the BMC Atrium Orchestrator Lightweight Activity Peer (LAP) 7.7.01 on the BMC Atrium Discovery appliance.

This is intended for executing long running operations, for example:
• Run a discovery and wait for completion
• Perform an offline compaction of the datastore

**To install BMC Atrium Orchestrator Lightweight Activity Peer**

Executing the graphical installer does not work as it requires X which is not available on a BMC Atrium Discovery appliance. Rather, you must perform a silent install. A silent install relies on a file containing all of the installation options that would normally be set using the GUI install. See the LAP installation worksheet for more information.

The installation consists of the following major steps:

• To create an installation options file
• To install BMC Atrium Orchestrator Lightweight Activity Peer
• Configure LAP to run on boot
• Timeout of long running processes
• Known issue with current BMC Atrium Discovery versions

**To create an installation options file**

Copy the following sample file into a text editor and edit the options to suit your environment. The fields are described in the BMC Atrium Orchestrator silent installation documentation.

```bash
-P installLocation=/usr/local/bmc/BAO/LAP
-J AO_WEBSERVER_PROTOCOL=https
-J AO_INSTALL_TYPE=install_new
-J AO_INSTALLING_FEATURES=LAP,WEBSERVER
-J AO_ENVIRONMENT_NAME=BAO Environment
-J AO_GRID_NAME=MyGrid
-J AO_GRID_TYPE=dev
-J AO_GRID_LOGGING_LEVEL=debug
-J AO_START_SERVER_ON_SUCCESS=true
-J AO_OCP_DEPLOYMENT_CONTEXT=baocdp
-J AO_GRAPH_DEPLOYMENT_CONTEXT=baograph
-J AO_PEER_NAME=CDP
-J AO_PEER_NET_CONFIG_CDP_CONTEXT=baocdp
-J AO_PEER_NET_CONFIG_CDP_HOST=baohost.bmc.com
-J AO_PEER_NET_CONFIG_CDP_PORT=38080
-J AO_REPOSITORY_PORT=28080
-J AO_REPOSITORY_USER_NAME=admin
-J AO_REPOSITORY_PASSWORD=
-J AO_PEER_NET_CONFIG_GRID_NAME=MyGrid
-J AO_PEER_NET_CONFIG_PEER_NAME=CDP
-J AO_PEER_NET_CONFIG_NEW_PEER_NAME=LAP
```
-J AO_PEER_NET_CONFIG_PROTOCOL=https  
-J AO_SECURITY_ACTIVE=false  
-J AO_SECURITY_WEB_PROTOCOL=https  
-J AO_WEBSERVER_PORT=38080  
-J AO_WEBSERVER_SHUTDOWN_PORT=38050  
-J AO_PEER_COMM_PORT=61719  
-J AO_WEBSERVER_HOST=addm.bmc.com

Save this file onto the BMC Atrium Discovery appliance file system, for example, /usr/tideway/tmp/optionsFile.

To install BMC Atrium Orchestrator Lightweight Activity Peer

To proceed with the install, create the install directory as root and change its ownership to the tideway user:

1. Log in to the appliance command line as the tideway user.
2. Run screen. Enter:
   
   [tideway@localhost ~]$ screen

3. Become the root user. Enter:

   [tideway@localhost ~]$ su
   
   Password:
   
   [root@localhost tideway]#

4. Create the installation directory and change its ownership to the tideway user. Enter:

   [root@localhost tideway]# mkdir /usr/local/bmc
   [root@localhost tideway]# chown tideway:tideway /usr/local/bmc
   [root@localhost tideway]# exit
   [tideway@localhost tideway]$

5. Open port 61721, or whichever port you intend to use for LAP communications. Use a text editor to add the following entries to the firewall configuration file /etc/sysconfig/iptables.

   :in_TIDEWAY_bao - [0:0]
   
   -A in_TIDEWAY_bao -j in_TIDEWAY_bao
   
   -A in_TIDEWAY_bao -p tcp -m tcp --sport 1024:65535 --dport 61721 -m state --state NEW,ESTABLISHED -j ACCEPT

6. Switch back to the tideway user:

   [root@localhost tideway]# exit
   [tideway@localhost tideway]$

7. Unset the Tomcat environment variables. Enter:

   [tideway@localhost tideway]# unset CATALINA_BASE CATALINA_HOME

8. Run the installer in silent mode as the tideway user, passing it the options file:

   [tideway@localhost tideway]$ sh linux_bao_server_installer_7_7_01.sh -i silent -DOPTIONS_FILE="/usr/tideway/tmp/optionsFile"

   Initializing blind certificate trust...
Composing URL from components: protocol=https, host=lon-vm-w2k12r2-bao.tideway.com, port=38080, file=baocdp/ws/install
Composed URL: [https://baohost.bmc.com:38080/baocdp/ws/install]
Creating config fetcher from [https://baohost.bmc.com:38080/baocdp/ws/install]
Composing URL from components: protocol=https, host=baohost.bmc.com, port=38080, file=baocdp
Composed URL: [https://baohost.bmc.com:38080/baocdp]
Composing URL from components: protocol=https, host=baohost.bmc.com, port=38080, file=baocdp/ws/install
Composed URL: [https://baohost.bmc.com:38080/baocdp/ws/install]
Creating config fetcher from [https://baohost.bmc.com:38080/baocdp/ws/install]
Apr 11, 2014 5:03:59 PM com.bmc.install.product.ao.utils.LocalCommandLineHelper
WARNING: LOG EVENT {Description=[Certificate stored in file <temp.cer>]} Apr 11, 2014 5:04:01 PM com.bmc.install.product.ao.utils.LocalCommandLineHelper
WARNING: LOG EVENT {Description=[Certificate was added to keystore]}

9. Check that you have the following java process running for the LAP:
/usr/local/bmc/BAO/LAP/jvm/bin/java -Djava.util.logging.config.file...

Configure LAP to run on boot

To configure LAP to run at boot time, use a text editor to edit the BMC Atrium Orchestrator start up bao.sh file.

1. Edit /usr/local/bmc/BAO/LAP/bin/bao.sh and set RUN_AS_USER=tideway
2. Remove the "-" from all of the "su -" commands in that file. This prevents the superuser inheriting the tideway user's environment variables. For example, the tideway user's Tomcat environment variables refer to the BMC Atrium Discovery Tomcat instance.
3. Become the root user. Enter:
[tideway@localhost ~]$ su
Password:
[root@localhost tideway]#
4. Create a symlink to the BAO startup script from the system startup scripts directory. Enter:
[root@localhost tideway]# ln -s /usr/local/bmc/BAO/LAP/bin/bao.sh /etc/init.d/bao_lap
[root@localhost tideway]# chkconfig bao_lap on [root@localhost
4.

1. Create an XML file called `tuning-config.xml` using the following code:

   ```xml
   <config>
     <adapter-manager>
       <adapter-call-timeout>36000000</adapter-call-timeout>
       <!-- timeout in milliseconds. 36000000 is 10 hours -->
       <!-- must exceed the longest expected execution time of a process -->
     </adapter-manager>
   </config>
   
   Copy the `tuning-config.xml` file to the following locations:
   
   - On the BMC Atrium Discovery appliance -- `/usr/local/bmc/BAO/LAP/config/`
   - On the BMC Atrium Orchestrator CDP host -- `C:\Program Files\BMC Software\BAO\CDP\config` or `/opt/bmc/BAO/CDP/config`
   - On the machine running the Dev Studio GUI (if required) -- `C:\Program Files\BMC Software\BAO\Studio\config`

Known issue with current BMC Atrium Discovery versions

The firewall changes will need to be applied again if BMC Atrium Discovery is upgraded.

**BMC Atrium Orchestrator VMotion events**

BMC Atrium Orchestrator integrates, automates, and orchestrates processes across multiple applications and tools, as well as across multiple IT groups, such as support and operations.

The integration between BMC Atrium Discovery and BMC Atrium Orchestrator currently supports a single as well as multiple VMotion events. When a running virtual machine is moved either manually (`VmMigratedEvent`), or by distributed resource scheduler (DRS) (`DrsVMMigratedEvent`) using VMware VMotion, this is detected by BMC Atrium Orchestrator. When integrated with BMC Atrium Discovery, BMC Atrium Orchestrator triggers a rescan of the virtual machine that was moved, and the source and destination hosts.

The following screen illustrates a rescan of two IP addresses that was automatically triggered by BMC Atrium Orchestrator:
This screen illustrates a rescan of two IP addresses that was automatically triggered by BMC Atrium Orchestrator.

Requirements

To detect when a running virtual machine is moved from one physical host to another.

1. Discovery must be running.
2. The physical hosts and the virtual host must already have been scanned by BMC Atrium Discovery.

If these requirements are not fulfilled, the VMotion event is not stored for BMC Atrium Discovery to catch up. However, this will occur as part of routine discovery.

⚠️ Non-matching hostnames

BMC Atrium Orchestrator derives hostnames from the VMware API and BMC Atrium Discovery derives hostnames by using the discovery techniques. The derived hostnames could either be a Fully Qualified Domain Name (FQDN), or an actual host name. For successful integration between BMC Atrium Orchestrator and BMC Atrium Discovery, the derived hostnames must match (that is, in both the products, the derived hostname for a specific host must be either be an FQDN, or an actual hostname). If there is a hostname mismatch, Discovery rescan is not triggered for that particular host.

Configuration in BMC Atrium Orchestrator

BMC Atrium Orchestrator must have the Discovery-SA-Synchronization runbook installed.

Use the BMC Atrium Orchestrator Grid Manager to specify the correct BMC Atrium Discovery appliance, username, and password.

1. From the Manage tab, click Modules.
2. From the Modules in Repository panel, select the Discovery-SA-Synchronization module.
3. Click Activate.
4. From the Activated on Grid panel, click the Discovery-SA-Synchronization module.
   The Edit Module Configuration page is displayed.
5. Select Virtualization-SA-Management.
6. Expand the Configuration tree to expand the following:
   Configuration VM Discovery
   For example:
This screen illustrates the Configuration tree expanded.

7. Enter values for the following items appropriate to your system.

| HTTP_POST_URL       | a. Click on the HTTP_POST_URL link.  
|                     | b. For the Value (string) enter https://appliancename/ui/aoint where appliancename is the name of your BMC Atrium Discovery appliance. This URL must use https.  
|                     | c. To save the changes, click OK.  
| password            | a. Click on the password link.  
|                     | b. For the Value (string) field, enter the password corresponding to the username specified below.  
|                     | c. To save the changes, click OK.  
| username            | a. Click on the username field.  
|                     | b. For the Value (string) field, enter the username on the BMC Atrium Discovery system with permissions to launch scans, that is, a member of the Discovery group. For more information, see Group Permissions.  
|                     | c. To save the changes, click OK.  


✅ Recommendation

Create a dedicated user to simplify security, support, and auditing.

8. To exit from BMC Atrium Orchestrator, click Logout.
Administering

The following topics provide information that is required to manage and maintain BMC Atrium Discovery:

- Navigating the administration interface
- Managing users and security
- Maintaining the appliance
- Managing standard data
- Managing clusters
- Consolidation
- Logging in to the appliance command line
- Using command line utilities

Navigating the administration interface

The Administration page provides access to many commonly required setup operations. To access the Administration page, click Administration on the primary navigation bar. View-only tabs are also provided which enable you to view the current configuration of appliance identification, support information, software, hardware, network interfaces and routing.

Common administration operations

The Administration page is divided into the following sections:

- **Discovery**— provides links to the configuration and management pages for Discovery. The available links are:
  - Platforms: configure the commands used for each target OS. This is described in Managing the discovery platform scripts.
  - SNMP Devices: view the network devices which can be discovered by BMC Atrium Discovery. This is described in Viewing an SNMP managed device.
  - Storage Devices: view the WBEM queries used to perform direct discovery of storage devices. See Storage device direct discovery.
  - Sensitive Data Filters: mask any sensitive data which might otherwise be seen in the command output. See Masking sensitive data.
  - Discovery Configuration: configure the Discovery process. This is described in Configuring discovery.
  - Discovery Consolidation: configure consolidation of discovery data between appliances. This is described in Consolidation.
  - Ciscoworks Import: import switch data from CiscoWorks. This is described in Importing network device data.
  - Vault Management: manage the Credential Vault. This is described in Managing the credential vault.
• **Device Capture:** capture an SNMP device using BMC Atrium Discovery to dump the MIB of an SNMP agent, which is then used to request that support be included in BMC Atrium Discovery for that SNMP device. This is described in *Capturing SNMP devices.*

• **Security** — provides links for the pages used to configure the security of the appliance and users. For more information, see *Managing users and security.*
  
  • Users: add, remove and modify users. This is described in *Managing system users.*
  
  • Groups: add, remove and modify groups. This is described in *Managing groups.*
  
  • Security Policy: configure the appliance security options. These are described in *Managing security policies.*
  
  • HTTPS: configure the HTTPS settings for the appliance. This is described in *Configuring HTTPS settings.*
  
  • Appliance Certificates: view and manage the certificates that the appliance uses for authentication.
  
  • LDAP: view information about integrating BMC Atrium Discovery with LDAP. This is described in *Managing LDAP.*
  
  • Single Sign On: Integrate BMC Atrium Discovery with *BMC Atrium Single Sign-On* (SSO) to simplify the user authentication. BMC Atrium Discovery supports other methods of *web authentication* using plugins to authenticate user logins, though *BMC Atrium SSO* is the preferred method.
  
  • Active Sessions: view all users who are currently logged in. This is described in *Viewing active sessions.*
  
  • Audit: configure the appliance audit feature. This is described in *Auditing the appliance.*

• **Appliance** — provides links for the configuration pages for the appliance. The pages related to the initial set up of the appliance are described later in this section. These are:
  
  • Configuration: configure and view appliance *identification,* viewing the appliance *specification* and *network information,* *mail settings,* and *JVM settings.*
  
  • Control: restart services, reboot, or shutdown the appliance, or place it into maintenance mode. If accessed from a cluster member perform the operations on a cluster. This is described in *Maintaining the appliance.*
  
  • Logs: manage logs and log levels. This is described in *Contents of the logs.*
  
  • Backup & Restore: operate the appliance backup feature that enables you to back up the appliance. This is described in *Backing up and restoring the appliance.*
  
  • Baseline status: check and configure the appliance baseline. This is described in *Baseline configuration.*
  
  • Performance: view charts showing the appliance performance over the last 30 days. This is described in *Monitoring appliance performance.*
  
  • Appliance Support: create an archive of diagnostic information for customer support. This is described in *Collecting additional data for support cases.*
  
  • Miscellaneous: configure miscellaneous settings.
  
  • JDBC Drivers: configure JDBC drivers that are available for Database Discovery and for Export. This is described in *Uploading new JDBC drivers.*
• Disk Configuration: enables you to move BMC Atrium Discovery data onto and create swap space on new disks on your appliance.
• Time Synchronization: configure time synchronization with an NTP server on your network or the internet.
• Cluster Management: create and configure clusters and perform cluster management tasks.
• Upgrade: upgrade a standalone appliance or cluster.

• **Model**— provides links to the pages used to configure and maintain the model, and import and export data.
  • View Taxonomy: view the system taxonomy. This is described in Viewing the system taxonomy.
  • Model Maintenance: configure model maintenance settings. This is described in Configuring model maintenance settings.
  • Custom Categories: set up data categories. This is described in Setting Up Standard Data Categories.
  • CMDB Sync: configure scheduled export synchronizations with BMC Atrium CMDB. This is described in Preparing BMC Atrium CMDB for synchronization.
  • Export: Perform export functions. This is described in Exporting data.
  • HRD Import: import Hardware Reference Data. This is described in HRD Import.
  • CSV Import: import generic data in CSV format. This is described in CSV Import.
  • Application Mapping Import: import application mapping definitions. This is described in import application mapping definitions.
  • Search Management: view any search in progress, and depending on your privileges, cancel searches. This is described in Using the Search service.
  • Channels: manage existing channels and create new channels. This is described in Managing channels.

### Managing users and security

This section explains how to set up and change system users and system groups, including how to create users, change passwords and security settings, and set up groups.

• Managing system users
• Managing groups
• Managing security policies
• Secure deployment
• Managing LDAP
• Integrating with BMC Atrium Single Sign-On
• Configuring Web authentication settings
• Viewing active sessions
• Auditing the appliance
Managing system users

The BMC Atrium Discovery Administrator is responsible for setting up details of all the users who are permitted to use the BMC Atrium Discovery system. Users are allocated a user name and a password, which they must enter in order to log in to the system. Each user is a member of one or more user groups, which defines the parts of the system that user is permitted to access. For example, users defined as members of the Admin group are able to create and edit user details, while members of the Public group cannot access these areas.

BMC Atrium Discovery can integrate with your corporate LDAP infrastructure. LDAP groups can be mapped to BMC Atrium Discovery groups and hence assigned permissions on the system. For information about setting up LDAP, see Managing LDAP.

As well as being the means of controlling user security, a user is actually set up on the system as a Person data object, and can subsequently be associated with other objects.

All actions on the system are recorded against a user’s ID for audit purposes. Users should always use their own ID and keep their security details safe.

Enabling other users

Typically, when you want to enable additional users to access BMC Atrium Discovery, you should log in as a user with the appropriate privileges. In addition to the system user, the following additional users are configured in BMC Atrium Discovery, but are not enabled by default:

- admin
- appmodel
- discovery

Use the system user only for configuration tasks which require system privileges.

To enable other users

1. Click the Administration tab.
2. In the Security Policy section of the Administration page, ensure the "Disabled Accounts can be reactivated” option is set to Yes
3. In the Security section of the Administration page, click the Users icon.
4. For each user, click the Set Password link.
5. On the Set Password page, enter the new password in each text entry field and click Apply.
6. After you have changed the passwords for each user, log off from the system user account by clicking the logout icon at the top right of the page.

Creating a new user

The BMC Atrium Discovery Administrator can set up new users and assign them to groups. Before creating users, you must ensure that you have set up all the groups that you need. See Managing groups.
To create a new user

1. From the Users page, click Add at the bottom of the page.
2. In the Add User page, enter the following details for the new user:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>Login ID of the user.</td>
</tr>
<tr>
<td>Full Name</td>
<td>Full name of user.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the password to be allocated to this user. Repeat the password for security reasons.</td>
</tr>
<tr>
<td>Password Rules</td>
<td>A read only display of the rules which are used to validate the password strength.</td>
</tr>
<tr>
<td>Groups</td>
<td>Select one or more groups that this user will be a member of. By default, all new users are members of the public group.</td>
</tr>
</tbody>
</table>

3. To save the changes, click OK.

⚠️ User names are case sensitive. That is, user names with the same spelling but different case are permitted, for example, Johnson and JOHNSON are not recognized as duplicates.

Amending a user’s details

You can change a user's name and the groups that they are a member of. The access defined by the group membership will apply the next time this user logs on.

To amend a user's details

1. From the Users page, click Edit next to the user.
   The page is redisplayed showing editable fields.
2. Amend or overwrite Full Name field.
3. Select one or more Groups that this user is to be a member of.
4. To save the changes, click OK.

Changing a user's password

If users forget their passwords or if a password is not kept secure, you can assign a new password.
To set a new password for a user

1. From the Users page, click **Set Password**.
   The page is redisplayed, showing blank Password fields. The existing password is not displayed.
   If the password policy requires a password to be changed, the label "MUST be changed" is displayed next to the user.
2. Enter a new password for this user in the **Password** field. Confirm the password in the **Verify Password** field.
3. To save the changes, click **Apply**. The new password will apply the next time the user attempts to log on.
   You can also specify that the user changes their password on their next login. To do this, click **Must Change Password**.

This section describes the preferred way to set or reset user passwords. However, you can also change users passwords at the command line.

**To reset the BMC Atrium Discovery user password at the command line**

The `tw_passwd` utility enables you to change the password of a specified user interface user. To use the utility, enter the following command at command prompt:

```
tw_passwd username
```

where:

- `username` is the name of the UI user to change.

For example:

```
[tideway@DE-32 ~]$ tw_passwd fred
New password:
Retype password:
Password set for user 'fred'.
[tideway@DE-32 ~]$
```

⚠️ **Changing passwords for command line users**

The `tw_passwd` utility is for changing UI users' passwords. To change the passwords for command line users, as the root user, use the Linux command `passwd`. This is described in **Changing the root and user passwords**
Reactivating a user account

If a user’s account is not used for a specified period of time, their account is deactivated. See Managing security policies for information on configuring account deactivation. To reactivate a deactivated user account you must be logged in as a member of the unlocker group and reactivating user accounts must be enabled in the security options page. You can also deactivate a user’s account manually. A deactivated account is never automatically reactivated.

To reactivate a locked user account

- Check that account reactivation is allowed (see Managing security policies)
- From the Users page, click Reactivate next to the user account to be reactivated.

Deleting a user

You can delete any existing user except for yourself or the default system-created users.

To delete an existing user

- From the Users page, click Delete next to the user to be deleted.

User permissions

User permissions in BMC Atrium Discovery are additive. When you grant a user an additional permission (through adding the user to another group), that permission is added to the user’s existing permissions. For example, if you grant appmodel permissions to a user with discovery permissions, the user gains no additional permissions because all of the appmodel permissions were already granted in the discovery permission set. Similarly, you cannot add readonly permissions to a system user in the hope of achieving a read only system user.

Managing groups

All users of the BMC Atrium Discovery system must be a member of one or more groups. Membership of groups defines the various BMC Atrium Discovery modules that a user is entitled to access. For example, users defined as members of the System group are able to create and edit user details, while members of the Public group cannot access these areas.

To log in, a user must be in a group that has permissions security/user/passwd, appserver/login and appserver/module/home. Only four default groups have this permission: readonly, public, system and admin. Every user must be a member of one of these four groups, or a member of a custom group that has at least these permissions.

For example, a user who is only in the discovery group cannot login. You should put a user that requires access to discovery commands into the discovery and public groups. The BMC Atrium Discovery Administrator is responsible for setting up details of all the user groups in the BMC Atrium Discovery system.
Each group is a collection of permissions. Permissions control granular access to BMC Atrium Discovery modules and are described in Group Permissions.

The default security groups

The default user groups and their security access rights are as follows:

- **admin** — These users have the highest level of customer access to the system.
- **appmodel** — These users can write and edit patterns, and create nodes to model business applications. They cannot view credentials but can run discovery (in order to test patterns).
- **discovery** — These users have access to all of the discovery-related data. They can start and stop discovery, add and remove credentials, and enable or disable audit logging.
- **cmdb-export-administrator** — These users have access to all of the export-related data. They can build, modify, delete and run Exporters.
- **public** — These users have read/write access to all of the system although they cannot access the discovery credentials.
- **readonly** — These users have read only access to the system. They cannot view the credentials for logging into target hosts.
- **system** — These users have full access to the system.
- **unlocker** — These users are able to unlock and unblock user accounts which have been locked or blocked after exceeding the number of permitted authentication failures. See Managing security policies for more information.

Listing all current groups

1. Click **Administration**.
2. From the Security section, click **Groups**.
   The Groups page lists all the current groups and allows you to edit details, delete groups or create a new group.

To create a new group

1. From the Groups page, click **Add** at the bottom of the page.
   The Add Group page is displayed. The page is arranged into functional areas, and then subdivided into columns. The arrangement of the columns from left to right is as follows:
   - Wildcard: contain items which when checked, select a number of permissions. When you mouseover a wildcard permission, it and the permissions it applies are highlighted.
   - Read: read permissions relating to the functional area.
   - Write: write permissions relating to the functional area.
   - Misc: miscellaneous permissions relating to the functional area, such as appliance reboot.
2. In **Group name**, enter a name for the new group.
3. Select the check boxes that indicate the BMC Atrium Discovery modules that members of this user group are allowed to access. The * wildcard matches anything, so selecting this check box will give unrestricted access to everything in the system.

4. to save the changes, click OK.

Once the group is set up you can add users. See Managing system users.

To amend group details

You can change a group name and the modules that group members can access. The access defined by the group membership will apply the next time users in this group log in.

1. From the Groups page, click Edit next to the user.

   The page is redisplayed showing editable fields.

2. Amend or overwrite the Name field.

3. Select one or more check boxes corresponding with the BMC Atrium Discovery modules that members of this group can access.

4. To save the changes, click OK.

To delete a group

You can delete any group provided you have created it initially. You cannot delete either the public or the system groups.

1. From the Groups page, click Delete next to the group to be deleted.

   The group is deleted and the system does not display any confirmation.

Group permissions

The following table shows the permissions assigned by default to each group in BMC Atrium Discovery. The individual permissions are described in System Group Permissions by Category.

<table>
<thead>
<tr>
<th>Group Name</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin</td>
<td>*</td>
</tr>
<tr>
<td>appmodel</td>
<td>admin/category/createmodify</td>
</tr>
<tr>
<td></td>
<td>admin/log/info</td>
</tr>
<tr>
<td></td>
<td>admin/log/read</td>
</tr>
<tr>
<td></td>
<td>reasoning/events/read</td>
</tr>
<tr>
<td></td>
<td>reasoning/events/write</td>
</tr>
<tr>
<td></td>
<td>reasoning/pattern/config</td>
</tr>
<tr>
<td></td>
<td>reasoning/pattern/execute</td>
</tr>
<tr>
<td></td>
<td>reasoning/pattern/write</td>
</tr>
<tr>
<td></td>
<td>reasoning/ranges/read</td>
</tr>
<tr>
<td></td>
<td>reasoning/ranges/write</td>
</tr>
<tr>
<td></td>
<td>reasoning/start</td>
</tr>
<tr>
<td></td>
<td>reasoning/startstop</td>
</tr>
<tr>
<td></td>
<td>reasoning/stop</td>
</tr>
<tr>
<td></td>
<td>ui/report/admin</td>
</tr>
<tr>
<td></td>
<td>vault/open</td>
</tr>
<tr>
<td>Group Name</td>
<td>Permissions</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------</td>
</tr>
</tbody>
</table>
| cmdb-export-administrator | cmdb_sync  
                      | vault/close  
                      | vault/credential_types/read  
                      | vault/credential_types/write  
                      | vault/credentials/read  
                      | vault/credentials/write  
                      | vault/open  |
| discovery          | consolidation/consolidation/write  
                      | consolidation/discovery/write  
                      | consolidation/read  
                      | discovery/credentials/test  
                      | discovery/filters/read  
                      | discovery/filters/write  
                      | discovery/kslave/read  
                      | discovery/kslave/write  
                      | discovery/options/read  
                      | discovery/options/write  
                      | discovery/platforms/read  
                      | discovery/platforms/write  
                      | discovery/port/settings  
                      | reasoning/events/read  
                      | reasoning/events/write  
                      | reasoning/pattern/config  
                      | reasoning/pattern/write  
                      | reasoning/ranges/read  
                      | reasoning/ranges/write  
                      | reasoning/start  
                      | reasoning/startstop  
                      | reasoning/stop  
                      | vault/credential_types/read  
                      | vault/credential_types/write  
                      | vault/credentials/read  
                      | vault/credentials/write  
                      | vault/open  |
| maintenance        | appliance/maintenance  |
| public              | appserver/login  
                      | appserver/module/*  
                      | model/datastore/main/read  
                      | model/datastore/main/write  
                      | model/datastore/partition/Audit/read  
                      | model/datastore/partition/Conjecture/read  
                      | model/datastore/partition/Conjecture/write  
                      | model/datastore/partition/DDD/read  
                      | model/datastore/partition/DDD/write  
                      | model/datastore/partition/Default/read  
                      | model/datastore/partition/Default/write  
                      | model/datastore/partition/Taxonomy/read  
                      | model/datastore/partition/_System/read  
                      | model/datastore/partition/_System/write  
                      | model/taxonomy/nodekind/read  
<pre><code>                  | model/taxonomy/relkind/read  |
</code></pre>
<table>
<thead>
<tr>
<th>Group Name</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/taxonomy/rolekind/read</td>
<td>reasoning/events/read</td>
</tr>
<tr>
<td>reasonings/events/write</td>
<td>reasoning/status</td>
</tr>
<tr>
<td>reports/read</td>
<td>reports/write</td>
</tr>
<tr>
<td>security/user/passwd</td>
<td>ui/dashboard/admin</td>
</tr>
<tr>
<td>ui/report/admin</td>
<td>ui/taxonomy/admin</td>
</tr>
<tr>
<td>readonly</td>
<td>appserver/login</td>
</tr>
<tr>
<td>appserver/module/*</td>
<td>model/datastore/main/read</td>
</tr>
<tr>
<td>model/datastore/partition/Audit/read</td>
<td>model/datastore/partition/Conjecture/read</td>
</tr>
<tr>
<td>model/datastore/partition/DDD/read</td>
<td>model/datastore/partition/Default/read</td>
</tr>
<tr>
<td>model/datastore/partition/Taxonomy/read</td>
<td>model/datastore/partition/_System/read</td>
</tr>
<tr>
<td>model/datastore/partition/_System/write</td>
<td>model/taxonomy/nodekind/read</td>
</tr>
<tr>
<td>model/taxonomy/relkind/read</td>
<td>model/taxonomy/rolekind/read</td>
</tr>
<tr>
<td>reports/read</td>
<td>reasoning/status</td>
</tr>
<tr>
<td>security/user/passwd</td>
<td>ui/report/admin</td>
</tr>
<tr>
<td>system</td>
<td>*</td>
</tr>
<tr>
<td>unlocker</td>
<td>security/group/read</td>
</tr>
<tr>
<td>security/options/read</td>
<td>security/user/activate</td>
</tr>
<tr>
<td>security/user/read</td>
<td></td>
</tr>
</tbody>
</table>

**System group permissions by category**

The system group security permissions are shown by category in the following tables.

⚠️ There are no permissions that restrict access to patterns. All logged in users can view patterns.

**Security permissions**

The following table shows the current group permissions relating to the security operations.
<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
</table>
| `security/group/read` | Enables the user to view and configure group membership to a user.  
  - `security/group/read`: View the user group names and the corresponding permissions.  
  - `security/group/write`: View, edit, delete, and create new groups.  
  You can manage user groups from the Groups page of the UI. To navigate to the page:  
  1. Click **Administration**.  
  2. From the Security section, click **Groups**. |
| `security/https/admin` | Enables the user to configure the HTTPS settings, which include:  
  - generating server keys and certificate signing requests  
  - uploading and sign server certificates  
  - uploading or downloading a CA certificate bundle  
  - enabling and disabling HTTP or HTTPS web access to the appliance  
  You can manage the HTTPS configuration from the HTTPS Configuration page of the UI. To navigate to the page:  
  1. Click **Administration**.  
  2. From the Security section, click **HTTPS**.  
  For more information, see [Configuring HTTPS settings](#). |
| `security/options/read` | Enables the user to view and configure the security options which include accounts and passwords, login page, and UI security page.  
  - `security/options/read`: View the security options for the appliance.  
  - `security/options/write`: Configure the security options settings for the appliance.  
  You can manage the options from the Security Policy page of the UI. To navigate to the page:  
  1. Click **Administration**.  
  2. From the Security section, click **Security Policy**.  
  For more information, see the [Managing security policies](#) page. |
| `security/user/activate` | Enables the user to unlock and re-activate accounts for other users from the Users page of the UI. To navigate to the page:  
  1. Click **Administration**.  
  2. From the Security section, click **Users**.  
  For more information, see [Accounts and passwords](#). |
| `security/user/passwd` | Enables the user to change her or his own BMC Atrium Discovery password from the UI.  
  For more information, see changing your password. |
| `security/user/read` | Enables the user to view and configure the user security information related to system users, groups, security policies, HTTPS settings, LDAP, Web authentication settings, active sessions, appliance audit, and so on.  
  - `security/user/read`: View the user security information.  
  - `security/user/write`: Configure the user security information.  
  For more information, see the [Managing users and security](#) page. |
| `security/sessions/view` | Enables viewing the discovery session log file. |
Credential vault permissions

BMC Atrium Discovery stores all passwords used to access customer devices in a credential vault which can be secured. The contents of the vault can be encrypted and secured using a passphrase.

The following table shows the current group permissions relating to the vault operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>vault/open</td>
<td>Enables the user to open, close, and set the passphrase for the credential vault from the Vault Management page of the UI. To navigate to the page:</td>
</tr>
<tr>
<td>vault/close</td>
<td></td>
</tr>
<tr>
<td>vault/passphrase</td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Discovery section, click Vault Management.</td>
</tr>
<tr>
<td></td>
<td>All three permissions are required to use the Vault Management page.</td>
</tr>
<tr>
<td>vault/credential_types/read</td>
<td>Enables the user to manage the following types of credentials which are based on the system to access:</td>
</tr>
<tr>
<td></td>
<td>• Device: To log on to hosts running a Unix, Linux OS, or Windows OS, or any SNMP enabled device such as routers and switches.</td>
</tr>
<tr>
<td></td>
<td>• Database: To query databases.</td>
</tr>
<tr>
<td></td>
<td>• Middleware: To query middleware such as web and application servers, and so on.</td>
</tr>
<tr>
<td></td>
<td>• Management System: For vCenter, vSphere, and mainframe credentials.</td>
</tr>
<tr>
<td></td>
<td>You can view and manage the credentials types from the Credentials page of the UI. To navigate to the page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Discovery.</td>
</tr>
<tr>
<td></td>
<td>2. Click Credentials.</td>
</tr>
<tr>
<td></td>
<td>For more information, see the Credentials page.</td>
</tr>
<tr>
<td>vault/credentials/read</td>
<td>Enables the user to view and manage credentials (For example, Windows proxies, vSphere credentials, and so on).</td>
</tr>
<tr>
<td></td>
<td>• vault/credentials/read: View the credentials.</td>
</tr>
<tr>
<td></td>
<td>• vault/credentials/write: Manage the credentials.</td>
</tr>
<tr>
<td></td>
<td>You can view and manage credentials from the Credentials page of the UI:</td>
</tr>
<tr>
<td></td>
<td>1. Click Discovery.</td>
</tr>
<tr>
<td></td>
<td>2. Click Credentials.</td>
</tr>
<tr>
<td></td>
<td>For more information, see the Credentials page.</td>
</tr>
<tr>
<td>vault/credentials/write</td>
<td>Enables the user to view and manage credentials (For example, Windows proxies, vSphere credentials, and so on).</td>
</tr>
</tbody>
</table>

Discovery permissions

The following table shows the current group permissions relating to the discovery operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>discovery/options/read</td>
<td>Enables the user to read the discovery options. These are separate from the main system settings.</td>
</tr>
<tr>
<td>discovery/options/write</td>
<td></td>
</tr>
<tr>
<td>discovery/credentials/test</td>
<td></td>
</tr>
</tbody>
</table>
Permission | Definition
--- | ---
| Enables the user to test discovery credentials. For example, from the UI, you can test:  
• Discovery credentials from the **Credential Tests** tab. For more information, see **Testing credentials**.  
• Mainframe credentials from the **Mainframe Credential Tests** tab. For more information, see **Testing mainframe credentials**.  

discovery/platforms/read | Enables the user to view and amend the platform discovery commands from the Discovery Platforms page.  
• discovery/platforms/read: View the platform discovery commands.  
• discovery/platforms/write: Amend the platform discovery commands.  
You can view and amend the platform discovery commands from the Discovery Platforms page of the UI. To navigate to the page:
1. Click **Administration**.  
2. From the Discovery section, click **Discovery Platforms**.  
   For more information, see Managing the discovery platform scripts.

discovery/host/access | Enables the user to query a host on the network. For more information, see **Query Builder**.

discovery/filters/read | Enables the user to view and modify sensitive data filters from the Sensitive Data Filters page of the UI. To navigate to the page:
1. Click **Administration**.  
2. From the Discovery section, click **Sensitive Data Filters**.  
   For more information, see **Masking sensitive data**.

discovery/kslave/read | Enables the user to view and modify the Windows proxies.  
You can manage Windows proxies from the Windows proxy management page of the UI. To navigate to the page:
1. Click **Discovery**.  
2. Click **Credentials**.  
3. Click **Windows Proxies**.  
   The Device Credentials page for Windows proxies (Windows proxy management page) is displayed.  
   For more information, see Managing Windows proxies.

discovery/kslave/write | Enables the user to view and modify the Windows proxies.  
You can manage Windows proxies from the Windows proxy management page of the UI. To navigate to the page:
1. Click **Administration**.  
2. From the Discovery section, click **Discovery Configuration**.  
   For more information, see port settings.

discovery/port/settings | Enables the user to configure the port settings that Discovery uses.  
You can manage the port settings from the Discovery Configuration page on the UI. To navigate to the page:
1. Click **Administration**.  
2. From the Discovery section, click **Discovery Configuration**.  
   For more information, see port settings.

**Consolidation permissions**

The following table shows the current group permissions relating to configuring consolidation and scanning appliances.
1. Permission

**consolidation/write**

Enables the user to change the configuration on the consolidation appliance (set as consolidation appliance and approve scanning appliances). You can manage consolidation from the Discovery Consolidation page on the UI. To navigate to the page:

1. Click Administration.
2. From the Discovery section, select **Discovery Consolidation**.

For more information about consolidation, see the **Consolidation** page.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>consolidation/write</td>
<td>Enables the user to add new consolidation targets to a scanning appliance from the Discovery Consolidation page on the UI.</td>
</tr>
<tr>
<td>consolidation/read</td>
<td>Enables the user to view the consolidation setup page from the Discovery Consolidation page on the UI.</td>
</tr>
</tbody>
</table>

### Datastore permissions

These permissions are a subsystem of the model. The following table shows the current group permissions relating to the datastore operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/datastore/main/read</td>
<td>Enables the user to read or write the datastore through the main user interface (UI).</td>
</tr>
<tr>
<td>model/datastore/main/write</td>
<td></td>
</tr>
<tr>
<td>model/datastore/partition/*/read</td>
<td>Enables the user to read or write to any partition which support user interaction. For more information, see the <strong>Partitions and History</strong> page.</td>
</tr>
<tr>
<td>model/datastore/partition/*/write</td>
<td></td>
</tr>
<tr>
<td>model/datastore/partition/name/read</td>
<td>Enables the user to read or write to the given partition. The name is one of:</td>
</tr>
<tr>
<td>model/datastore/partition/name/write</td>
<td>• Audit</td>
</tr>
<tr>
<td></td>
<td>• Conjecture</td>
</tr>
<tr>
<td></td>
<td>• DDD</td>
</tr>
<tr>
<td></td>
<td>• Default</td>
</tr>
<tr>
<td></td>
<td>• Taxonomy</td>
</tr>
<tr>
<td></td>
<td>• _System</td>
</tr>
<tr>
<td></td>
<td>For more information, see the <strong>Partitions and History</strong> page.</td>
</tr>
<tr>
<td>model/datastore/internal/cluster</td>
<td>An internal permission. Do not use this.</td>
</tr>
</tbody>
</table>

### Audit permissions

These permissions are a subsystem of the model. The following table shows the current group permissions relating to the audit operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/audit/read</td>
<td>Enables the user to read the audit log. Audit logs are stored in /usr/tideway/log. The files can be accessed directly from the appliance command line or in the log viewer from the UI. Logs can be downloaded from the appliance through the Support Services administration page.</td>
</tr>
<tr>
<td>model/audit/write</td>
<td>Enables the user to write to the audit log.</td>
</tr>
</tbody>
</table>
### Permission Definition

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
</table>
| `model/audit/purge` | Enables the user to purge the audit log. You can purge the audit log of all events that are over one month old (events less than one month old cannot be deleted) from the Audit Purge page. To navigate to the Audit Purge page:  
  1. Click **Administration**.  
  2. From the Security section, click **Audit**.  
  3. Click **Purge**.  
     For more information, see [purging audit logs](#). |
| `model/audit/admin` | Enables the user to administer the audit service. You can configure the reporting on audit events from the Audit Logs page. To navigate to the Audit Logs page:  
  1. Click **Administration**.  
  2. From the Security section, click **Audit**.  
  3. Click **Audit Logs**.  
     For more information, see the [Auditing the appliance](#) page. |

### Reasoning permissions

These permissions are a subsystem of the model. The following table shows the current group permissions relating to the reasoning operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>reasoning/start</td>
<td>Enables the user to start reasoning. You can start reasoning by using the <code>tw_scan_control</code> utility. For more information, see <code>tw_scan_control</code>.</td>
</tr>
<tr>
<td>reasoning/startstop</td>
<td>Enables the user to start and stop reasoning. You can start and stop reasoning by using the <code>tw_scan_control</code> utility. For more information, see <code>tw_scan_control</code>.</td>
</tr>
<tr>
<td>reasoning/stop</td>
<td>Enables the user to stop reasoning. You can stop reasoning by using the <code>tw_scan_control</code> utility. For more information, see <code>tw_scan_control</code>.</td>
</tr>
<tr>
<td>reasoning/status</td>
<td>Enables the user to view the reasoning status information. You can view the reasoning status by using the <code>tw_reasoningstatus</code> utility. Typically this utility is used by Customer Support as a troubleshooting tool for investigating possible problems. For more information, see <code>tw_reasoningstatus</code>.</td>
</tr>
<tr>
<td>reasoning/ranges/read</td>
<td>Enables the user to view the Discovery Status page. The status of the discovery process displays on the Home tab in the Discovery Status summary. This page also displays the current status of the reasoning process. For more information, see <a href="#">Viewing discovery status</a>.</td>
</tr>
</tbody>
</table>
| reasoning/ranges/write | Enables the user to cancel consolidations or local scans. You can cancel consolidation or local scans from the Discovery Status page. To navigate to the Discovery Status page:  
  1. Click Discovery.  
  2. Click Discovery Status.  
     For more information, see [Viewing discovery status](#). |
| reasoning/events/read | An internal permission. Do not use this. |
1. Permission  

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>reasoning/events/write</td>
<td>An internal permission. Do not use this.</td>
</tr>
<tr>
<td>reasoning/internal</td>
<td>Enables the user to configure patterns.</td>
</tr>
<tr>
<td>reasoning/pattern/config</td>
<td>Enables the user to configure patterns.</td>
</tr>
<tr>
<td>reasoning/pattern/execute</td>
<td>Enables the user to execute patterns.</td>
</tr>
</tbody>
</table>
| reasoning/pattern/write     | Enables the user to write patterns using pattern templates from the appliance.  

For more information, see Pattern Configuration.

2. Permission  

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/search/list</td>
<td>Enables the user to view searches submitted by all users.</td>
</tr>
<tr>
<td>model/search/cancel</td>
<td>Enables the user to cancel searches submitted by all users.</td>
</tr>
</tbody>
</table>

**Search permissions**

These permissions relate to listing and cancelling searches using the Search Management Page.

To navigate to the Search Management page:

1. Click Administration.
2. From the Model section, click Search Management.

For more information on viewing and cancelling searches, see Using the Search service.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/search/list</td>
<td>Enables the user to view searches submitted by all users.</td>
</tr>
<tr>
<td>model/search/cancel</td>
<td>Enables the user to cancel searches submitted by all users.</td>
</tr>
</tbody>
</table>

**Taxonomy permissions**

These permissions are a subsystem of the model. The following table shows the current group permissions relating to the taxonomy operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>model/taxonomy/nodekind/read</td>
<td>Enables the user to read node kind information.</td>
</tr>
<tr>
<td>model/taxonomy/nodekind/write</td>
<td>Enables the user to write node kind information.</td>
</tr>
<tr>
<td>model/taxonomy/relkind/read</td>
<td>Enables the user to read relationship kind information.</td>
</tr>
<tr>
<td>Permission</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>model/taxonomy/relkind/write</td>
<td>Enables the user to write relationship kind information.</td>
</tr>
<tr>
<td>model/taxonomy/rolekind/read</td>
<td>Enables the user to read role kind information.</td>
</tr>
<tr>
<td>model/taxonomy/rolekind/write</td>
<td>Enables the user to write role kind information.</td>
</tr>
</tbody>
</table>

**Application server permissions**

The following table shows the current group permissions relating to the application server operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>appserver/login</td>
<td>Enables the user to log in to the appserver.</td>
</tr>
<tr>
<td>appserver/debug</td>
<td>Enables the user to debug the appserver.</td>
</tr>
<tr>
<td>appserver/module/name</td>
<td>Enables the user to access the given module. The name is one of:</td>
</tr>
<tr>
<td></td>
<td>• Application</td>
</tr>
<tr>
<td></td>
<td>• Discovery</td>
</tr>
<tr>
<td></td>
<td>• Home</td>
</tr>
<tr>
<td></td>
<td>• Infrastructure</td>
</tr>
<tr>
<td></td>
<td>• Reports</td>
</tr>
<tr>
<td></td>
<td>• Setup</td>
</tr>
<tr>
<td></td>
<td>• System</td>
</tr>
<tr>
<td>appserver/module/*</td>
<td>Enables the user to access any module.</td>
</tr>
<tr>
<td>appserver/sessionaccess</td>
<td>The user is allowed to see sessions.</td>
</tr>
</tbody>
</table>

**Specific UI permissions**

The following table shows the current group permissions relating to specific user interface operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ui/dashboard /admin</td>
<td>Enables the user to administer the dashboard. You can use the <code>tw_config_dashboards</code> utility to configure and customize dashboards in BMC Atrium Discovery. For more information, see <code>tw_config_dashboards</code>.</td>
</tr>
<tr>
<td>ui/datastore /admin</td>
<td>Enables the user to administer the datastore.</td>
</tr>
<tr>
<td>ui/taxonomy /admin</td>
<td>Enables the user to administer the taxonomy.</td>
</tr>
</tbody>
</table>
| ui/report /admin         | Enables the user to access the Generic Search Query page and enter search queries. To navigate to the Generic Search Query page:  
  1. Click the **Search** icon to the left of the Search box at the top right of the User Interface. The Search Options in the drop down panel is displayed.  
  2. Click the **Generic Search Query** link. For more information, see **Using the Search service**. By default, all admin users get this permission. |
Appliance administration permissions

The following table shows the current group permissions relating to the appliance administration operations.

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
</table>
| admin/category/createmodify | Enables the user to create and modify categories from the Custom Categories page:
To navigate to the Custom Categories page:
1. Click Administration.
2. From the Model section, click Custom categories.
   For more information, see Setting up standard data categories. |
| appliance/info/read | Enables the user to view appliance information (identity, support information, read-only information about the appliance software and hardware configuration, and so on) from the Appliance Configuration page.
To navigate to the Appliance Configuration page:
1. Click Administration.
2. From the Appliance section, click Configuration.
   For more information, see Configuring appliance settings. |
| appliance/info/write | Enables the user to configure appliance information from the Appliance Configuration page.
To navigate to the Appliance Configuration page:
1. Click Administration.
2. From the Appliance section, click Configuration.
   For more information, see Configuring appliance settings. |
| appliance/maintenance | Enables the user to put the appliance into maintenance mode from the Appliance Control page. To navigate to the Appliance Control page:
1. Click Administration.
2. From the Appliance section, click Control.
   For more information, see Using maintenance mode. |
| appliance/reboot | Enables the user to reboot the appliance from the Appliance Control page. To navigate to the Appliance Control page:
1. Click Administration.
2. From the Appliance section, click Control.
   For more information, see Rebooting or shutting down the appliance. |
| appliance/reportsusage/reset | Enables the user to reset the report usage statistics. |
| appliance/restart | Enables the user to restart the appliance from the Appliance Control page. To navigate to the Appliance Control page:
1. Click Administration.
2. From the Appliance section, click Control.
   For more information, see Rebooting or shutting down the appliance. |
<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
</table>
| appliance/shutdown | Enables the user to shut down the appliance from the Appliance Control page. To navigate to the Appliance Control page:  
   1. Click **Administration**.  
   2. From the Appliance section, click **Control**.  
   For more information, see Rebooting or shutting down the appliance. |
| baseline/admin | Enables the user to change the baseline configuration (such as the recipients of automatic emails, and the messages to be included) from the Appliance Baseline page. To navigate to the Appliance Baseline page:  
   1. Click **Administration**.  
   2. From the Appliance section, click **Baseline Status**.  
   For more information, see **Baseline configuration**. |
| baseline/read | Enables the user to view the baseline configuration from the Appliance Baseline page. To navigate to the Appliance Baseline page:  
   1. Click **Administration**.  
   2. From the Appliance section, click **Baseline Status**.  
   For more information, see **Baseline configuration**. |
| baseline/update | Enables the user to update the baseline configuration after changes have been seen from the Appliance Baseline page. To navigate to the Appliance Baseline page:  
   1. Click **Administration**.  
   2. From the Appliance section, click **Baseline Status**.  
   For more information, see **Baseline configuration**. |
| cluster/file_distribution | An internal permission. Do not use this. |
| cluster/management | Enables the user to use cluster management operations from the Cluster Management page. To navigate to the Cluster Management page:  
   1. Click **Administration**.  
   2. From the Appliance section, click **Cluster Management**.  
   For more information, see **Managing clusters**. |
| cluster/monitored_operation | An internal permission. Do not use this. |
| admin/log/info | Enables the user to view log information.  
As each BMC Atrium Discovery component and script runs, it outputs logging information. Logs are all stored in `/usr/tideway/log`.  
You can access the files directly from the appliance command line or in the log viewer in the UI. |
| admin/log/read | Enables the user to read log files.  
Logs are all stored in `/usr/tideway/log`.  
You can access the files directly from the appliance command line or in the log viewer in the UI. |
<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin/log/delete</td>
<td>Enables the user to delete log files. You can delete the log files from the Logs page. To navigate to the Logs page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click Logs.</td>
</tr>
<tr>
<td></td>
<td>For more information, see view and delete logs.</td>
</tr>
<tr>
<td>admin/loglevel/read</td>
<td>Enables the user to read the appliance log level from the Logs page. To navigate to the Logs page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click Logs.</td>
</tr>
<tr>
<td></td>
<td>For more information, see view log levels.</td>
</tr>
<tr>
<td>admin/loglevel/write</td>
<td>Enables the user to change the service log levels at runtime from the Logs page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click Logs.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Changing log levels at runtime.</td>
</tr>
<tr>
<td>Import</td>
<td></td>
</tr>
<tr>
<td>admin/import/ciscoworks</td>
<td>Enables the user to import data using the CiscoWorks importer.</td>
</tr>
<tr>
<td></td>
<td>You can use the tw_imp_ciscoworks utility to import CiscoWorks network device data from the command line. For more information, see tw_imp_ciscoworks.</td>
</tr>
<tr>
<td>admin/import/csv</td>
<td>Enables the user to import CSV data from the Import CSV Data page. To navigate to the CSV Data page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Model section, click CSV Import.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Importing CSV data.</td>
</tr>
<tr>
<td>admin/import/hrd</td>
<td>Enables the user to import Hardware Reference data (HRD) from the Import Hardware Reference Data page. To navigate to the Hardware Reference Data page:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Model section, click HRD Import.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Importing Hardware Reference Data.</td>
</tr>
<tr>
<td>Interface</td>
<td></td>
</tr>
<tr>
<td>admin/interface/read</td>
<td>Enables the user to view interface information from the Appliance Configuration page for network interfaces. To navigate to the Appliance Configuration page for network interfaces:</td>
</tr>
<tr>
<td></td>
<td>1. Click Administration.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click Configuration.</td>
</tr>
<tr>
<td></td>
<td>3. Click Network Interfaces.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Viewing network interface and routing settings.</td>
</tr>
<tr>
<td>admin/interface/write</td>
<td>This permission is not used.</td>
</tr>
<tr>
<td>Routing</td>
<td></td>
</tr>
<tr>
<td>Permission</td>
<td>Definition</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>admin/routing/read</td>
<td>Obsolete permission.</td>
</tr>
<tr>
<td>admin/routing/write</td>
<td>Obsolete permission.</td>
</tr>
</tbody>
</table>

**DNS**

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin/dns/read</td>
<td>Enables the user to read DNS information. You can view the DNS (Name Resolution) information from the Appliance Configuration page for name resolution. To navigate to the page:</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Administration</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click <strong>Configuration</strong>.</td>
</tr>
<tr>
<td></td>
<td>3. Click <strong>Name Resolution</strong>. For more information, see Configuring name resolution settings.</td>
</tr>
<tr>
<td>admin/dns/write</td>
<td>The user is allowed to write DNS information. You can configure the DNS (Name Resolution) information from the Appliance Configuration page for name resolution. To navigate to the page:</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Administration</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click <strong>Configuration</strong>.</td>
</tr>
<tr>
<td></td>
<td>3. Click <strong>Name Resolution</strong>. For more information, see Configuring name resolution settings.</td>
</tr>
</tbody>
</table>

**Email Configuration**

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>admin/mail/read</td>
<td>Enables the user to view email configuration information from the Appliance Configuration page for mail settings. To navigate to the page:</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Administration</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click <strong>Configuration</strong>.</td>
</tr>
<tr>
<td></td>
<td>3. Click <strong>Mail Settings</strong>. For more information, see Configuring mail settings.</td>
</tr>
<tr>
<td>admin/mail/write</td>
<td>Enables the user to configure the mail settings from the Appliance Configuration page for mail settings. To navigate to the page:</td>
</tr>
<tr>
<td></td>
<td>1. Click <strong>Administration</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. From the Appliance section, click <strong>Configuration</strong>.</td>
</tr>
<tr>
<td></td>
<td>3. Click <strong>Mail Settings</strong>. For more information, see Configuring mail settings.</td>
</tr>
</tbody>
</table>

**System**

<table>
<thead>
<tr>
<th>Permission</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>system/configuration /read</td>
<td>Enables the user to read system configuration from the command line utilities and from the UI.</td>
</tr>
<tr>
<td>system/configuration /write</td>
<td>Enables the user to write system configuration from the command line utilities and from the UI.</td>
</tr>
<tr>
<td>system/settings/read</td>
<td>Enables the user to read system settings from the command line utilities and from the UI.</td>
</tr>
<tr>
<td>system/settings/write</td>
<td>Enables the user to write system settings from the command line utilities and from the UI.</td>
</tr>
</tbody>
</table>
The 'all' permission (*) allows the user to perform any tasks in BMC Atrium Discovery. Each user has a token which is assigned by the security system and whenever a privilege is requested by a user, the security service checks the database to see if that particular user has permission to carry out that particular task.

However, the first check that BMC Atrium Discovery carries out is to see if the user has the * permission. If the answer is yes, no further privilege checks will be carried out.

Managing security policies

Many organizations enforce security policies on user access to their systems. BMC Atrium Discovery supports this by providing configurable security options and multiple authentication mechanisms. You can configure the following:

- Accounts and Passwords
  - Password strength and expiry
  - Forced password change
  - Account blocking after authentication failures
  - Deactivation of unused accounts
- Login Page
  - Appearance of the login page
  - Legal banner messages
  - Allow browser autocomplete
- UI Security page
  - Prevent Cross Site Framing

Configuring these settings is described in the following sections.

Accounts and passwords

To configure the security options:

1. Click Administration.
2. From the Security section, click Security Policy.

The options on the Security Policy page are described below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Blocking</td>
<td>User accounts can be blocked after a number of unsuccessful login attempts. Select the number of attempts from the drop-down list. Choose from the following 1, 2, 3, 4, or 5 attempts. If you do not want accounts to be blocked, select Never. The default is 3.</td>
</tr>
<tr>
<td>Automatically Unblock</td>
<td>After a user account is blocked, it can be automatically unblocked after a specified period. Select the period from the drop-down list. Choose from the following 1, 2, 3, 4, 5, 10, 15, 20, 30, or 60 minutes. If you do not want accounts to be automatically unblocked, select Never. The default is 10 minutes. If you select Never, there is a chance that you could lock out the system account. See #Blocking of the System Account for more information.</td>
</tr>
</tbody>
</table>
### Field Name | Details
---|---
Account Deactivation | Unused user accounts can be deactivated after a specified period of time. Select the period from the drop-down list. Choose from the following 15, 30, 45, 60, 75, 90, 105, and 120 days. If you do not want accounts to be deactivated, select Never. The default is that disabled accounts cannot be reactivated.

Disabled Accounts can be reactivated | Select Yes or No to allow user accounts to be reactivated. You will need an administrator to reactivate the account.

Minimum Password Length | You can specify a minimum length for passwords. Select a minimum length from the drop-down list. Choose a length from 1 to 32 characters. Select None to enforce no minimum length. The default is 8 characters.

Password History | You can specify a password history length to prevent users from recycling passwords too quickly. Select the password history length from the drop-down list. Choose from 3, 5, 10, or 20. Select None to enforce no restrictions on password reuse. The default is 10.

Password constraints | Select from the following check boxes to apply constraints to the password contents. In general, the password quality improves with more selected check boxes:
- Must contain uppercase characters — for example AIV. The default is true.
- Must contain lowercase characters — for example aiv. The default is true.
- Must contain numeric characters — for example 174. The default is true.
- Must contain special characters — for example ^£). The default is true.
- Must not contain sequences — for example AAA, ppp, or 222. The default is true.

Password Expiry Period | You can specify a maximum length of time for passwords before they are automatically expired. Select an expiry period from the drop-down list. Choose from 30, 45, 60, 75, 90, 105, and 120 days. Select None to enforce no expiry period. The default is 90 days.

Password Expiry Warning | Users can be warned that their password will expire soon when login into the user interface. The warning icon is displayed in the Navigating the BMC Atrium Discovery UI#Dynamic toolbox. Select a warning period from the drop-down list. Choose from 5, 10, and 15 days. Select Never to give users no warning of an expiring password. The default is 10 days. The expiry warning cannot be set to more than the expiry period.

### Blocking of the system account

In the following scenario, the system user account can be locked.

- Account blocking is enabled (the default).
- Automatic account unblocking is disabled (not the default).
- A user repeatedly attempts to log in unsuccessfully to the UI as the system user. An administrator is required to log in to the system and unblock the account.

### Login page

You can configure the appearance of the login page and add a legal notice to the login page.

To configure the login page:

1. From the Security section of the Administration tab, select Login Page Options.
   The options on the The Security Options: Login page are described below:
Plain login page
Where security is a concern, you can choose to remove all banners and logos from the login page. Doing so reduces the risk of attack by hiding the nature of system from a would be attacker. Select Yes to do this. This option is not available in the BMC Atrium Discovery Community Edition.

The BMC favicon (shown in browser tabs) remains visible when you use the plain login page. If you want to remove the favicon, you should rename the /var/www/html/favico.ico file. For example favico.ico.hidden.

Legal Notice
Enter an additional legal notice in the Legal Notice text field.

Allow Browser Autocomplete
You can specify whether to allow data stored in browsers to be used to autocomplete fields in the UI. Select Yes or No.

UI security page
You can prevent cross site framing to defend against possible "clickjacking" attacks.
To configure the UI security page:

1. From the Security section of the Administration tab, select UI Security.

Secure deployment
Communication between components of BMC Atrium Discovery is secured using TLS, and authenticated using certificates. The communication is:

- Between appliance and Windows proxy
- Between scanning appliance and consolidator
- Between members of a cluster

For information about securing communication between the user's browser and the appliance web user interface, see Configuring HTTPS settings.

When one part of BMC Atrium Discovery connects to another, each end of the connection checks that the other end is approved. The certificate can be thought of as the public key half of a public-private key pair — if the endpoint's private key details match the expected public certificate details, the connection is approved. Note that although the underlying protocols are the same, the trust model in use in BMC Atrium Discovery is different to the trust model involved in HTTPS used on the public web. Technical details are given below.
Certificates are small text files containing an ugly jumble of letters and numbers. To enable visual comparison of certificates, they are reduced to a **fingerprint**, which is a somewhat shorter sequence of ugly hexadecimal digits, for example '7D:0B:13:54:4B:F1:39:87:A0:AB:4E:FA:3D:A0:A2:0C:2F:0A:A1:BA'. Certificate fingerprints are shown in various parts of BMC Atrium Discovery so that the identities of the endpoints can be confirmed.

**Maximum security**

The most secure way to deploy the components of BMC Atrium Discovery is for each component to have its own unique key and certificate. This is the default scheme for new deployments. In this scheme, each connection between components must be approved; any connections that have not been approved are denied.

In versions of BMC Atrium Discovery prior to 10.1, a single default certificate was used in all deployments. That default certificate is referred to as the **legacy certificate**. For added security, the recommendation was to deploy custom keys signed by a private certificate authority. Upon upgrade from an earlier version, the existing keys and certificates are retained, so that existing communication links are not broken. To increase security of existing deployments, we recommend that once all parts of the deployment have been upgraded to version 10.1 or later, unique keys and certificates are generated for each component.

**Approving connections**

When connecting components of BMC Atrium Discovery, the components' certificates must be exchanged. The actual certificate exchange is performed automatically, but the connection must be confirmed via the user interfaces on both ends of the connection.

- Between **scanner and consolidator**, the scanner connects to the consolidator and the connection must be approved on the consolidator.
- Between the BMC Atrium Discovery appliance and the **Windows Proxy**, the link can be established from either end, and must be approved at the other end.
- In a BMC Atrium Discovery **cluster**, the cluster is considered a single component and all members share the same key and certificate details. When a new member joins the cluster, it automatically receives the key and certificate details.

Once a connection has been established, if either end changes its key and certificate details, the communication link will fail, and must be re-approved.

**Upgrade considerations**

Upon upgrade from a release prior to 10.1 the existing keys and certificates are retained, so that existing communication links continue to function. To increase the overall security of the system we recommend that once all parts of the system have been upgraded to 10.1 or later, the old keys are replaced by new unique ones for each component. Components can be upgraded one at a time — there is no need to switch all components from legacy keys at the same time.
If the upgraded appliance or Windows proxy was previously using the default legacy keys, then when the switch to unique keys is made, appliances automatically approve the new certificate. If the default keys had been replaced with custom site-specific keys, a switch to new unique keys will disable the existing communication links until they are re-approved. When an appliance key and certificate is changed, the Windows proxy always requires approval of the new certificate.

If it is necessary to add a newly installed appliance or Windows proxy to a system involving older version components, communication will not immediately succeed because the newly installed component will have unique keys that are not known to the old components. In this situation, the newly-installed component should be switched to use the legacy key and certificate via the certificate management interface.

Managing keys and certificates

Within the BMC Atrium Discovery user interface, details of the appliance’s own key and certificate, and the certificates of other components that it trusts are visible within the Administration tab, in Security > Appliance Certificates. The page permits the generation of new keys, and the installation of the legacy keys for successful interaction with older-version components.

On the Windows proxy, access information about the proxy’s own key and certificate in the Proxy Manager via Edit > Key and Certificate Management. As with the Appliance Certificates page, the dialog permits the generation of a new key and certificate and the installation of the legacy key. Certificate details of the connected appliances are available via Edit > Known Appliances.

Technical details

The TLS (Transport Layer Security) protocols used in BMC Atrium Discovery make use of both public-key and symmetric-key cryptography, and secure hashing. This section explains the effects of these cryptography techniques, how they are used in TLS, and how BMC Atrium Discovery uses the features of TLS to achieve authentication, authorization and end-to-end encryption.

- In symmetric-key cryptography, a message is encrypted with a key, and can only be decrypted with the same key.
- In public-key cryptography, a pair of keys is used. A message that has been encrypted with the private key can only be decrypted with the public key; conversely, a message that has been encrypted with the public key can only be decrypted with the private key.
- Secure hashing takes a message and reduces it to a short hash value. Changing the message changes the hash, and it is computationally infeasible to generate another message with the same hash value.

⚠️ Unless switching from the legacy keys to unique ones after an upgrade, generating a new key and certificate will break all existing communication links to other components, until the links are re-established.
As a simple example of how these techniques can be used, suppose Alice wishes to communicate with Bob. They each have a public-private key pair, and each knows the other's public key. Public-key encryption algorithms are computationally expensive, so they wish to use a symmetric-key algorithm to exchange messages. The interaction is as follows:

1. Alice invents a new key to use with a symmetric-key algorithm. This is the session key.
2. Alice encrypts the session key first with her private key, and then with Bob's public key. She sends the doubly-encrypted message to Bob.
3. Bob decrypts the message with first his private key and then Alice's public key. Only Bob can do that because only he possesses the correct private key. He is sure that the message really came from Alice because her public key can only decrypt messages encrypted with her private key.
4. Alice and Bob communicate for as long as necessary using the exchanged session key.

TLS negotiates communication in a similar manner, except that it adds an extra layer of indirection that is useful on the public web. Suppose that Alice and Bob do not know each others' public keys, but they do both trust Charlie, and they know his public key. Charlie can act as a certificate authority (or CA). Alice and Bob each ask Charlie to sign their public keys:

1. Alice sends her public key to Charlie.
2. Charlie uses a secure hash algorithm to generate a hash of the public key.
3. Charlie encrypts the hash with his private key. This is the key signature.
4. He sends the combination of public key and encrypted hash back to Alice. Now the public key has been signed, it is called a certificate.

Since Charlie's public key is public, anyone can confirm that Alice's certificate was signed by Charlie, because they can decrypt the encrypted hash using his public key and check that the hash value is correct. Bob can similarly have his public key signed by Charlie or another trusted certificate authority.

Now, Alice and Bob can negotiate a session key to communicate as before, with an additional interaction at the start:

1. Alice sends her certificate to Bob.
2. Bob checks that the certificate was signed by a trusted certificate authority (Charlie).
3. Seeing that it was, he now trusts that the public key does indeed belong to Alice, so he sends Alice his certificate.
4. Alice can also check that Bob's certificate was signed by Charlie, or any other certificate authority she trusts.

Alice and Bob now have each others' public keys, so they can proceed as before. This is, in a somewhat simplified form, how TLS works.
TLS on the public web

On the public web, a number of companies have established themselves as certificate authorities. Browser manufacturers have chosen to accept these certificate authorities, and embed their public keys within the browsers. Web sites that wish to use HTTPS pay a certificate authority to sign their public keys. Now, when Alice wishes to use her web browser to connect to Bob's Bank Inc's web site, her browser can see that the Bob's Bank Inc certificate was signed by Charlie's Certificate Authority Ltd, and it is happy to proceed. End users on the web do not have certificates signed by a certificate authority, so Alice's browser does not have a signed certificate, and the Bob's Bank server does not attempt to verify one — the TLS protocol is configured to only authenticate the server-side certificate, and not require a client certificate. Hopefully Bob's Bank uses another mechanism to verify that Alice really is who she claims to be.

The trust model on the web is only intended to allow browsers to verify that web servers are who they claim to be. It relies on there being a small number of public certificate authorities that are trusted by all browsers.

TLS in BMC Atrium Discovery

The way TLS is used in BMC Atrium Discovery is different to the public web. Both client and server wish to authenticate each other, and all connections are individually approved so there is no need to have a single trusted certificate authority. The TLS protocols require the use of certificate authorities, however, so when each BMC Atrium Discovery component (appliance or Windows proxy) generates a unique public-private key pair for itself, it also generates a unique certificate authority key pair for itself, and signs its own certificate with it. The certificate authority public key is also signed with its own private key, so it is referred to as a self-signed CA Certificate. After generation, the system discards the CA's private key so it can never sign any more certificates with that CA.

Each component in BMC Atrium Discovery therefore has a unique single-use CA with its own CA certificate. It is these CA certificates that are exchanged between components, and must be approved for communication to be permitted.

Configuring HTTPS settings

The HTTPS Configuration page enables you to configure the HTTPS settings for the appliance. This includes:

- Generating server keys and certificate signing requests
- Uploading and signing server certificates
- Upload a CA certificate bundle to the appliance, or download them from the appliance
- Upload a Certificate Revocation List to revoke access to the appliance
- Enable and disable HTTP or HTTPS web access to the appliance

To access the HTTPS Configuration page, select HTTPS from the Security section of the Administration tab. The server key displays the private key for the appliance.
If BMC Atrium Discovery is integrated with a Web Authentication (Single Sign On) solution, you need to replace a default Certificate Authority (CA) bundle on BMC Atrium Discovery.

To generate a server key
On the Server Key tab of the HTTPS Configuration page, the existing key details are shown, or if no key exists, empty fields are displayed.

1. To generate a server key, enter relevant information in the editable fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>A read-only description of the current server key status. For example, this might contain information on the length and modification date of the key in use.</td>
</tr>
<tr>
<td>Server Name</td>
<td>An editable field automatically populated with the hostname of the standalone appliance. If the appliance is a cluster member, it is the cluster alias, or if an alias has not been set then the cluster name is used.</td>
</tr>
<tr>
<td>Country Code</td>
<td>The two character country code for the country in which the appliance is located, for example GB.</td>
</tr>
<tr>
<td>State or Province</td>
<td>The state or province in which the appliance is located, for example Yorkshire.</td>
</tr>
<tr>
<td>Locality</td>
<td>The locality in which the appliance is located, for example York.</td>
</tr>
<tr>
<td>Company Name</td>
<td>The company name, for example, BMC Software.</td>
</tr>
<tr>
<td>Department</td>
<td>The department using the appliance. This field is optional.</td>
</tr>
<tr>
<td>Email Address</td>
<td>The email contact for users of this appliance. This field is optional.</td>
</tr>
<tr>
<td>RSA key length</td>
<td>The RSA key length. Select one of the following from the drop down list: 1024, 2048, or 4096 bits.</td>
</tr>
</tbody>
</table>

⚠️ The values in the Server Key tab must match those used by the certificate authority.

2. When you have entered the required information, click Generate New Server Key. The new server key is saved as $TIDEWAY/etc/https/server.key onto the appliance’s file system. A certificate signing request is also generated, it is called server.csr and is saved in the same location.

When you have a key and a signing request, it must be signed before it can be used. You can do this using one of the following methods:

- Use a certificate authority: continue with this procedure.
- Sign the certificate yourself: see Self Signing a Server Certificate.

3. To download the certificate signing request, click Download CSR. Use the download dialog to choose the location on your local filesystem in which to save the file.
4. Send the certificate signing request file to your certificate signing authority for signing. When the certificate signing authority has approved the request, they will generate the corresponding certificate and return it as a .crt file.

### Uploading a server certificate

1. When your certificate signing authority has approved the request, they will return a certificate. Save this file on your local filesystem.
2. On the HTTPS Configuration page, click the **Server Certificate** tab.
3. Click **Browse** next to **Certificate File**: and select the server certificate you saved in Step 1 of this procedure.
4. Click **Upload New Certificate**.
   The new certificate is uploaded onto the appliance.

### Self signing a server certificate

If you do not use a certificate authority, but still require HTTPS access to the appliance, you can use the self-signing feature.

To self sign a certificate:

1. Ensure that you have created a server key and certificate signing request on the appliance using the procedure described in to generate a server key.
2. In the HTTPS Configuration page, click **Server Certificate => Self Sign**.
   The server key that you created is signed and saved as a new certificate called server.crt.
3. Enable HTTPS access. See Enabling or disabling HTTP and HTTPS access to the appliance for more information.
   When you access BMC Atrium Discovery using HTTPS, you will be prompted to accept the certificate once per each session.

### Uploading or downloading a CA certificate bundle

The CA certificate bundle that is included by default contains a number of certificates from public certificate authorities. These are usually known as Trusted root certificates, or Trusted Intermediate Certificates. You can continue to use these or replace them with a certificate bundle from a certificate authority used by your organization. Your system administrator should tell you whether to use the supplied bundle or will provide you with one supported by your organization.

⚠️ If you do not have a CA bundle, either the default supplied with the appliance, or one supplied by your organization, you will be unable to use HTTPS.
The default CA bundle is stored on the appliance in the following directory:
/etc/pki/tls/certs/ca-bundle.crt
When the certificate signing authority has approved the request, they will generate the corresponding certificate bundle and return it as a .crt file.
To replace the certificate bundle with one from a certificate authority used by your organization:

1. On the HTTPS Configuration page, click **CA Certificates**.
2. Click **Browse** next to **CA Certificate Bundle File** and select the server certificate you saved in Step 1 of this procedure.
3. Click **Upload New CA Certificate Bundle**.
   The new certificate bundle is uploaded.

To download the existing CA certificate bundle:

1. Click **Download CA Certificate Bundle**.
2. Use the download dialog to choose the location on your local filesystem in which to save the file.

**Using a Certificate Revocation List to revoke access to the appliance**

You can use a Certificate Revocation List (CRL) to ensure that certificates that have been revoked by the CA can no longer be used to access the appliance. A CRL contains a list of certificates which have been revoked by the CA. You can also add compromised certificates to the CRL.

To apply a CRL

1. On the HTTPS Configuration page, click **Certificate Revocation List**.
2. Click **Browse** next to **Certificate Revocation List** and select the CRL to apply.
3. Click **Upload CRL**.
   The CRL is uploaded and applied.

**Enabling or disabling HTTP and HTTPS access to the appliance**

Use a two stage approach to enabling redirect to HTTPS. Configure the HTTPS and test that it is configured correctly and permits access to authenticated users. Only then should you enable redirect to HTTPS.

If HTTPS is not configured correctly, and you enable redirect to HTTPS, you could be locked out of the appliance.

By default users can access the BMC Atrium Discovery over HTTP. You can enable HTTPS connections on this page and specify that attempts to connect over HTTP should be redirected to HTTPS.

By default HTTP access is enabled and HTTPS access is disabled.
1. On the HTTPS Configuration page, click HTTPS tab. The following screen illustrates an example of HTTPS enabled and HTTP redirected to HTTPS:

![HTTPS Configuration Screen](image)

This screen illustrates an HTTPS enabled and HTTP redirected to HTTPS.

- To enable HTTPS access, from the HTTPS list, select **Enabled**.
- To disable HTTPS access, from the HTTPS list, select **Disabled**.
- To enable HTTP access, from the HTTPS list, select **Enabled**.
- To redirect HTTP access attempts to HTTPS, from the HTTP list, select **Redirect to HTTPS**.

**Replacing a default Certificate Authority bundle**

The Certificate Authority (CA) bundle is required when using Web Authentication (SSO). You need to upload a list of PEM encoded X.509 public certificates for each root and intermediate Certificate Authority in the chain in order for clients to be successfully verified.

To replace a default CA:

1. In Administration tab, open **HTTPS Configuration**.
2. On the **CA Bundle** tab, choose file with the certificate you would like to use as CA and click **Upload New Bundle**.

   **Note:** If you are using Microsoft Certificate Services, you need to use the **Download CA certificate chain** option with Base 64 encoding. This produces a PKCS #7 encoded file (.p7b) that is converted to the correct format automatically during the upload.

**Appliance certificates**

Communication within BMC Atrium Discovery is secured using TLS, and authenticated using certificates.

On the Appliance Certificates section you can get an overview of the certificates trusted by the system, manage the certificates for this appliance, as well as view the certificates for the appliances and proxies this appliance communicates with.

On the Known Certificates section you can see the details of all the certificates registered with the appliance (Windows proxies, consolidated appliances, and so on).

- **Navigating to the Appliance Certificates**
- **Viewing the appliance key/appliance certificate**
- **Generating a new key and certificate**
- **Viewing portable CA certificate**
- **Viewing known certificates**
After an upgrade from BMC Atrium Discovery versions earlier than 10.1, you are alerted to replace the legacy key and certificate by generating a new ones.

For an appliance that is a part of a consolidation setup, when you generate a new key and certificates, the appliance immediately exchanges keys and certificates with other members of the consolidation setup. This happens only once, immediately after the first time the legacy key and certificate is replaced with the new ones.

For new installation of the BMC Atrium Discovery v10.1 a unique set of keys is generated automatically.

Navigating to the Appliance Certificates

To navigate to the Appliance Certificates:

1. Click on the Administration tab on the top bar.
2. Click the Appliance Certificates in the Security section.

Viewing the appliance key/appliance certificate

To view the appliance key or appliance certificate, click Show details link next to the Appliance Key or Appliance Certificate fingerprint.

The following example shows the Appliance Key details:
1. Generating a new key and certificate

If you believe that an appliance's key has been compromised, or you otherwise wish to disable all existing communication with an appliance, you can generate a new key and certificate.

**Notes**

Unless switching from the legacy keys to unique ones after an upgrade, generating a new key and certificate will break all existing communication links to other components, until the links are re-established manually.

To generate new key and certificate:

1. On the Appliance Certificates page, click **Install new certificate and key**.
2. When prompted, confirm the action by clicking **Yes**.

Installing legacy key and certificate
To integrate a newly-installed appliance into a system with earlier-version proxies or appliances, you can install the legacy key and certificate.

### Note

Switching to the legacy key and certificate will break any existing communication that was configured using the unique key and certificate. The links should be re-established manually.

To roll back to using the legacy key that is used in BMC Atrium Discovery version 10.0 and earlier, install legacy key and certificate:

1. On the Appliance Certificates page, click **Install legacy certificate and key**.
2. When prompted, confirm the action by clicking **Yes**.

**Viewing portable CA certificate**

On the Appliance Certificates page, click **Show portable CA Certificate**.

The CA Certificate dialog displays the CA certificate in a portable format (pem) that might be used for transferring the certificate to the appliance manually.
Viewing known certificates

To view a list of known certificates:

1. Click on the **Administration** tab on the top bar.
2. Click the **Appliance Certificates** in the Security section.

The information fields for a known certificate are arranged in the following groups:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Name</td>
<td>The name of the registered certificate, that usually consists of the type of the registered item and its name (for example, proxy: AD).</td>
</tr>
<tr>
<td>Show details/Hide details</td>
<td>The link that expands/collapses the certificate contents.</td>
</tr>
<tr>
<td>File</td>
<td>Name of the file on disc that stores the certificate (for example, proxy_AD.pem).</td>
</tr>
<tr>
<td>Fingerprint</td>
<td>The certificate fingerprint.</td>
</tr>
</tbody>
</table>
Managing LDAP

LDAP is commonly used to access user or group information in a corporate directory. Using your corporate LDAP infrastructure to authenticate users can reduce the number of administrative tasks that you need to perform in BMC Atrium Discovery. LDAP groups can be mapped to BMC Atrium Discovery groups and hence assigned permissions on the system. The way in which BMC Atrium Discovery integrates with your LDAP infrastructure depends on the schema that is implemented in your organization.

If you are using LDAP authentication there is no need to set up local user accounts for LDAP users on BMC Atrium Discovery.

- LDAP Terms
- Configuring LDAP
- LDAP group mapping
- Troubleshooting

LDAP Terms

The following terms are used in the sections describing BMC Atrium Discovery LDAP configuration:

- **Directory Information Tree (DIT)** — The overall tree structure of the data directory queried using the LDAP protocol. The structure is defined by the schema. Each entry in a directory is an object; one of two types:
  - **Containers** — A container is like a folder: it contains other containers or leaves.
  - **Leaves** — A leaf is an object at the end of a tree. Leaves cannot contain other objects.
- **Domain Component (dc)** — Each element of the Internet domain name of the company is given individually.
- **Organizational Unit (ou)** — Organizations in the company.
- **Common Name (cn)** — The name of a person.
- **Distinguished Name (dn)** — The complete name for a person, including the domain components, organisational unit, and common name.

An example Directory Information Tree is shown below.

```
dc=tideway,dc=com
  ou=engineering
    cn=Timothy Taylor
      telephoneNumber=1234
      email=t.taylor@bmc.com
    ou=test
```
The login procedure

When a user attempts to log in through the user interface, BMC Atrium Discovery first checks to see whether the username represents a local account. If no local account exists, and LDAP has been configured correctly, BMC Atrium Discovery attempts to authenticate against the directory and then performs an account lookup to return the group memberships of that account. If the group mappings have been enabled, and configured correctly, then authentication takes place and the user is logged in with the local BMC Atrium Discovery rights as defined in the group mapping.

The Global Catalog

The Global Catalog is a distributed data repository that contains a searchable, partial representation of every object in every domain in a multidomain Active Directory Domain Services (AD DS) forest. The global catalog is stored on domain controllers that have been designated as global catalog servers and is distributed through multimaster replication. Searches that are directed to the global catalog are faster because they do not involve referrals to different domain controllers.

Configuring LDAP

To configure the LDAP settings:

1. From the Security section of the Administration tab, select LDAP. The LDAP page is displayed showing the LDAP tab.

This graphic shows the main LDAP configuration screen.
The options on this page are described below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDAP Support</td>
<td>Select <strong>Enabled</strong> or <strong>Disabled</strong> to enable or disable LDAP support for this appliance.</td>
</tr>
</tbody>
</table>
| Connection Status| Displays a message regarding the status of the connection to the LDAP server. For example:  
  • No LDAP operations performed (last update: timestamp)  
  • Invalid credentials (last update: timestamp)  
  • Connection established (last update: timestamp)  
  • Can't contact LDAP server (last update: timestamp)  

| Server URI      | The address of the LDAP server to connect to. For example:  
  ldap://engineering.bmc.com:3268 or  
  ldaps://engineering.bmc.com  
  The default LDAP port is 389 and the default LDAPS port is 636.  
  For multiple (failover) LDAP servers, enter a space separated list of LDAP server URIs.  
  When using the Microsoft Active Directory group mode for LDAP, you can also use port 3268 to reference the Global Catalog. Check with your LDAP administrator to ensure that you use the correct port.                                                                                                     |
| LDAPS           | Displays a message regarding the CA certificate and provides controls enabling you to upload, remove or replace a certificate. Many large enterprises have their own CAs that will provide a root CA certificate which will allow the appliance to trust the LDAP server's certificate it receives over the network.  
  To upload a certificate, click the Browse button, select the new certificate using the file upload dialog, then click Apply.  
  To replace an existing CA Certificate, select the Remove CA Certificate check box and click Apply. When the page is refreshed, add the new certificate as above.  
  You should contact your organization's LDAP administrator to obtain a CA certificate. Multiple CA certificates can be uploaded by concatenating into a single file prior to upload.  
  You cannot delete a CA certificate when LDAPS is enabled. Likewise, you cannot enable LDAPS without a CA Certificate loaded. In both these cases you will encounter a Cannot use LDAPS without a CA Certificate warning.                                                                                     |
| Bind Username   | The user name with which to connect to the LDAP server. For example, user01@bmc.com.                                                                                                                                                                                                                                                                                                                                                           |
| Bind Password   | The password that corresponds to the user name entered in the **Bind Username** field. Check the box to modify the password.                                                                                                                                                                                                                                                                                                           |
| Bind Timeout    | The length of time that the appliance will wait before the login is assumed to have failed.                                                                                                                                                                                                                                                                                                                                          |
| Search Base     | The location in the directory from which the LDAP search begins. For example: dc=bmc,dc=com. This restricts the search to the bmc container in the directory information tree.                                                                                                                                                                                                                                                                              |
1. When you are not using group mapping (see #LDAP Group Mapping) any BMC Atrium Discovery group you enter, must be entered in lower case.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search Template</td>
<td>Specifies the template to use to search for the user name in the LDAP database. For example: (userPrincipalName=%(username)s) This queries the LDAP database for the userPrincipalName attribute which is equal to %(username)s, which is the user name string entered at the login prompt.</td>
</tr>
<tr>
<td>Search Timeout</td>
<td>If no response is received from the server in this length of time, the query times out. Select a timeout value from the drop-down list.</td>
</tr>
<tr>
<td>Search Scope</td>
<td>Defines how deep to search within the search base. &quot;Base&quot;, or zero level, indicates a search of the base object only. &quot;One level&quot; indicates a search of objects immediately subordinate to the base object, but does not include the base object itself. This is typically used to search for objects immediately contained in the search base level. &quot;Sub Tree&quot; indicates a search of the base object and the entire subtree of which the base object distinguished name is the topmost object. Select the required scope from the drop-down list.</td>
</tr>
<tr>
<td>User Cache Timeout</td>
<td>The appliance queries the LDAP server for user information and caches the results to avoid overloading the LDAP server. Select a timeout value from the drop-down list. Values less than 10 minutes are not recommended. You can also clear the cache manually using the Flush Cache button.</td>
</tr>
<tr>
<td>Group Cache Timeout</td>
<td>The appliance queries the LDAP server for group information and caches the results to avoid overloading the LDAP server. Select a timeout value from the drop-down list. Values less than 1 hour are not recommended. You can also clear the user and group cache manually using the Flush Cache button.</td>
</tr>
</tbody>
</table>
| Group Mode                  | The group mode determines the way that the LDAP server is queried for group information, it should match the LDAP server used by your organization. Select one of the following LDAP server types from the drop-down list:  
  - Microsoft Active Directory  
  - SunONE Directory Server  
  - Other |
| Group Attribute on User node| The LDAP attribute name to search for when running a group query. The attribute is on the User node, and provides a list of distinguished names of groups that the user belongs to. For example, the attribute might be called "memberOf" and contain data such as "cn=sales.ou=groups, dc=bmc,dc=com". This field is user editable when the Other Group Mode is selected from the Group Mode drop-down (if the User node does not contain such an attribute, this field should be empty so the Membership Attribute on Group node will be used instead). When any other mode is selected the field is automatically populated. |
| Group Query                 | The LDAP query that is used to find Group objects. It is usual to match the nodes' Object Class, for example: (objectclass=group). This field is user editable when the Other Group Mode is selected from the Group Mode drop-down. When any other mode is selected the field is automatically populated. |
| Membership Attribute on Group node | The LDAP attribute name to search for to determine whether an individual is a member of a group. The attribute is on the Group nodes, and provides a list of names of users. For example, the attribute might be called "member". This field is user editable when the Other Group Mode is selected from the Group Mode drop-down. When any other mode is selected the field is automatically populated. |

2. To save the LDAP settings, click **Apply**.
Configuring LDAP for use with BMC Atrium SSO

Depending on how your LDAP servers are configured, user authentication via Atrium SSO may work, but then user authorization in BMC Atrium Discovery fails. This occurs because Atrium SSO sends BMC Atrium Discovery the first part of the user’s DN as their userid.

For example, for a DN of the following format:

```
  dn: CN=ADDM QA. TEST,CN=Users,DC=addmsqa,DC=bmc,DC=com
```

The part that must be matched by the search that BMC Atrium Discovery runs is:

```
  ADDM QA. TEST
```

To do this, for the example above, set the Search Base to:

```
  cn=users,dc=addmsqa,dc=bmc,dc=com
```

and the Search Template to:

```
  (cn=%(username)s)
```

Changing from LDAPS to LDAP

When you reconfigure BMC Atrium Discovery to use LDAP when it was previously configured to use LDAPS, you must remove the CA Certificate, and change the URI in a single step otherwise you will encounter a Cannot use LDAPS without a CA Certificate warning. To do this:

1. Edit the URI to point to the LDAP server’s ldap:// URI. Do not click Apply button yet.
2. Select Remove CA Certificate.
3. Click Apply.

Changing from LDAP to LDAPS

When you reconfigure BMC Atrium Discovery to use LDAPS when it was previously configured to use LDAP, you must add a CA certificate before you attempt to enter an ldaps:// URI.
**LDAP group mapping**

The LDAP group mapping enables you to assign membership of BMC Atrium Discovery groups to LDAP groups. If you do not use group mapping, users will be only be assigned to groups in BMC Atrium Discovery which are exactly the same as the the LDAP groups that they are members of, that is, in LDAP form dc=tideway,dc=com,ou=engineering...

**To enable or disable LDAP group mapping**

1. From the LDAP page, select the **Group Mapping** tab.

   ![LDAP Groups tab](image)

   **This graphic shows the LDAP Groups tab.**

   The LDAP Group Mapping page lists the LDAP groups that are assigned to BMC Atrium Discovery security groups. For each LDAP group, the appliance security groups to which it is assigned are listed. Links for each action that you can perform are provided for each group.

2. Select **Enabled** or **Disabled** from the drop-down list to enable or disable LDAP group mapping.

**To add or edit LDAP Group Mapping starting from a username**

1. From the LDAP page, select the **Group Mapping** tab.

2. Click **Lookup User**.

3. In the LDAP User Lookup dialog enter the **Username** and click the **OK** button.

4. The system looks up the username in LDAP and displays the results.

   ![Lookup User results](image)

   **This graphic shows the Lookup User results screen.**
LDAP Groups: For each LDAP group of which the user is a member, displays existing group mappings and provides an add link or an edit link.

Mapped Groups: Displays the final list of mapped groups for this user.

Details: Displays whether the information was obtained from the local cache and the total number of groups to which this user belongs.

5. Click **Add** to create a new group mapping or **Edit** to modify an existing group mapping.

6. Select the appliance security groups to which you want to assign the LDAP group.

7. To save the mapping, click **Apply**.

**To add an LDAP Group Mapping starting from an LDAP group name**

1. From the LDAP page, select the **Group Mapping** tab.
2. Click **Add**.
3. On the Add LDAP Group Mapping page, enter a search term for the common name into the LDAP Group field and click the **Search** button. A list of matches is displayed. If more than ten entries match, the first ten are shown and a label is displayed at the bottom of the list showing how many additional matches there are.
4. Select the matching LDAP group from the list. The LDAP groups field is not case sensitive. All LDAP groups returned from the LDAP server are displayed in lower case.
5. Select the appliance security groups to which you want to assign the LDAP group.
6. To save the mapping, click **Apply**.

**To edit an LDAP Group Mapping starting from an LDAP group name**

1. From the LDAP page, select the **Group Mapping** tab.
2. For each LDAP group listed, an edit link and a delete link are provided.
3. Click **Edit**.
4. Select the appliance security groups to which you want to assign the LDAP group.
5. To save the mapping, click **Apply**.

**To delete an LDAP Group Mapping**

1. From the LDAP page, select the **Group Mapping** tab.
2. For each LDAP group listed, an edit link and a delete link are provided.
3. To remove an LDAP group mapping, click **Delete**.

**Troubleshooting**

If you receive a "Can't Contact LDAP Server error" in the Connection Status field, this might be caused by certificate problems rather than simple connectivity (wrong URI, port and so forth). Check that the certificate you are using is the one you received from your LDAP administrator.

If the login fails when attempting LDAP authentication, set the security log `/usr/tideway/log/tw_svc_security.log` level to debug.
Where the account used to bind to the directory fails to authenticate look for messages similar to the following:

```
-1285350512: 2010-08-13 10:00:46,843: security.authenticator.ldap: DEBUG: Attempt to auth
-1285350512: 2010-08-13 10:00:47,117: security.authenticator.ldap: DEBUG: LDAP passwd for

```

If you are using group mapping and are experiencing login failures, check that group mappings have been correctly defined for one or more LDAP groups to which the user belongs. See To add or edit LDAP Group Mapping starting from a username.

## Integrating with BMC Atrium Single Sign-On

BMC Atrium Single Sign-On (BMC Atrium SSO) is a mechanism that simplifies the user authentication process and enables you to:

- Integrate various BMC Software products for Single Sign-On user authentication.
- Authenticate with integrated BMC Software products by entering the login credentials only once.
- Log off integrated BMC Software products by logging off from one of those.

BMC Atrium Discovery supports BMC Atrium SSO, version 8.1 and later, for user authentication.

**Note**

The current version of BMC Atrium Discovery is shipped with the BMC Atrium Single Sign-On (SSO) 8.1 agent. The 8.1 agent is compatible with the recently released SSO 9.0 server. However, this earlier version of the agent does not support some of the latest features and enhancements of BMC Atrium SSO 9.0. Currently, upgrading the agent to 9.0 on BMC Atrium Discovery is not possible.

To configure BMC Atrium SSO user authentication, you must:

1. Have access to the BMC Atrium SSO server in your environment.
2. Register the BMC Atrium SSO Web Agent with the BMC Atrium SSO Server.
3. Activate the registered BMC Atrium SSO Web Agent.
4. Verify that the BMC Atrium SSO Web Agent is registered with the BMC Atrium SSO server.

## Accessing the BMC Atrium SSO Server

To configure BMC Atrium SSO user authentication, you must access the BMC Atrium SSO Admin Console. If BMC Atrium SSO is not already installed and configured in your environment, you can download the latest version from the BMC Electronic Product Distribution (EPD) site and install and configure it for user authentication.
The BMC Atrium SSO server URL must always be specified as a Fully Qualified Domain Name (FQDN).

To access to the BMC Atrium SSO Admin Console from the BMC Atrium SSO Server:

1. Click Start > All Programs > BMC Software > BMC Atrium SSO > Administrator.
2. Log in with the BMC Atrium SSO administrator password.

To access to the BMC Atrium SSO Admin Console from a client system:

1. Enter the BMC Atrium SSO URL, this must be an FQDN, including the port number, into the browser and press Return.
   For example, https://hostname.bmc.com:8443/atriumSSO
2. When prompted, enter the BMC Atrium SSO administrator credentials and log in.

Configuring the BMC Atrium SSO Web Agent with the BMC Atrium SSO Server

Before you configure the BMC Atrium SSO Web Agent with the BMC Atrium SSO Server, read the notes on Configuring LDAP for use with BMC Atrium SSO, ensure that the LDAP settings are configured and you are able to login to the BMC Atrium Discovery appliance as an LDAP user with administrative privileges (so that once you have activated the BMC Atrium SSO integration, you will be able to login again and change the configurations, if required).

To configure the BMC Atrium SSO Web Agent, you must perform the following from the BMC Atrium Discovery UI:

1. Register the web agent with the BMC Atrium SSO Server
2. Activate the web agent

To register the web agent

**Note**

If you plan to enable Federal Information Processing Standard (FIPS) Publication 140-2, you must do it before registering the web agent. The Atrium SSO Server must also be in FIPS 140-2 mode. For more information, see Running in FIPS compliant mode.

1. Log in to the BMC Atrium Discovery appliance UI as a user with administration privileges.
2. From the Security section of the Administration tab, click Single Sign On.
3. From the Registration section of the Atrium SSO tab, complete the following parameters:
   a. Atrium SSO Web Agent: (Read-only field) Displays whether the web agent is registered or not.
b. **Agent FQDN**: Enter the FQDN for the web agent. You must use an FQDN, or you may be unable to log in to the BMC Atrium Discovery UI. You cannot specify localhost.localdomain or .local FQDNs.

c. **Atrium SSO Server URL**: Enter the URL for the BMC Atrium SSO Server. You must use an FQDN, or you may be unable to log in to the BMC Atrium Discovery UI.

d. **Atrium SSO Realm**: Leave the BMC Atrium SSO Realm name as the default value of /BmcRealm.

e. **Admin Username**: Leave the BMC Atrium SSO administrator user name as the default value of amadmin.

f. **Admin Password**: Enter the BMC Atrium SSO administrator password.

4. To complete the registration, click **Register**.

The registration might take a few seconds and on completion a message, *Atrium SSO Web Agent registered with Atrium SSO Server*, is displayed on the UI. For registered web agents, it is not possible to edit the registration parameters. To edit the registration parameters you must deregister the agent. When you deregister a web agent, enter the BMC Atrium SSO administrator password in the Admin Password field and click **Deregister**.

**To activate the web agent**

1. From the Activation section of the Atrium SSO page, click **Activate**.

   Activating the web agent restarts the Apache Web Server in the background and might take a few seconds and on completion a message, *Please allow a few seconds for the changes to be applied. You may need to reauthenticate* is displayed along with a **Refresh** link.

2. Click **Refresh**.

3. If prompted, you must re-authenticate your BMC Atrium SSO session.

   The following fields are displayed:

   a. **Status**: Displays the activation status of the web agent. For activated web agents it is **Activated**. Otherwise, the status is **Deactivated**.

   b. **LDAP**: Displays whether LDAP support is enabled or not.

      If LDAP support is not configured, click on the corresponding link to complete the configuration.

   c. **HTTPS**: Displays whether HTTPS support is enabled or not.

      Configuring HTTPS support is highly recommended, and if not configured, click on the corresponding link to complete the configuration.

   d. **Restart Web Agent**: Enables you to restart the web agent which in turn restarts the Apache Web Server in the background. Typically, if you make any configuration changes in the BMC Atrium SSO Server, you must restart the web agent for the changes to take effect.

   e. **Deactivate**: Enables you to deactivate the web agent. Deactivating the web agent requires you to re-authenticate your BMC Atrium SSO session and may take a few seconds. Once deactivated, you are presented with the BMC Atrium Discovery appliance’s login UI.
BMC Atrium SSO configurations in a cluster

If you update the BMC Atrium SSO configuration in the coordinator, it are automatically updated in the existing members. However, if you add a new member, the configuration is not automatically updated as the BMC Atrium SSO administrator user ID and password are not stored in the appliance. You must manually apply the configurations on the newly added member.

To manually apply the BMC Atrium SSO configurations on a newly added member, perform the following on the coordinator:

1. Click Administration > Single Sign On.
2. From the Registration section of the Atrium SSO tab, enter the Admin Password.
3. Click Synchronize.
   The synchronization of the configurations on the new member might take a few seconds and a message, Atrium SSO configuration being synchronized, is displayed on the UI of the coordinator.

When the synchronization of the configuration is completed, the Atrium SSO Web Agent for the new member is registered. You must manually activate the web agent from the member's UI.

Verifying registration of the BMC Atrium SSO Web Agent with the BMC Atrium SSO Server

To verify the registration of the BMC Atrium SSO Web Agent with the BMC Atrium SSO Server:

1. From the BMC Atrium SSO Admin Console, click Agent Details.
   The Agent Manager displays the list of the web agents configured for Single Sign-On.
2. Verify if the corresponding web agent is available in the list or not. It may show as Down although it is working.
   If BMC Atrium SSO user authentication is configured, the list displays the corresponding web agent. However, if BMC Atrium SSO user authentication is not configured or the web agent has been deregistered, the list does not display the web agent.

When BMC Atrium SSO user authentication is successfully configured and you attempt to access the login UI for an integrated product, it redirects you to the BMC Atrium SSO login page. Once you get authenticated for the session, you can access all the integrated products without having the need to get authenticated for each of those for as long as the session is valid.

Configuring Web authentication settings

BMC Atrium Discovery supports a number of web authentication plugins. You can view and configure these on the Web Authentication Methods Page.

The following web authentication methods are supported:
SSL Client Certificate Verification: The client's SSL Certificate is verified by the web server. The user name is extracted from the certificate and used for authorization via LDAP. Requires LDAP support.

SSL Certificate Lookup: The user is authenticated by looking up custom parts of the client's SSL Certificate via LDAP. The certificate is not verified, but it must be valid. Requires LDAP support.

RSA SecurID Authentication: Authentication is performed by the RSA Authentication Agent. The username is used for authorization via LDAP. Requires HTTPS and LDAP support. This is available in BMC Atrium Discovery 9.0 SP1 and later.

HTTP Header: BMC Atrium Discovery is integrated with Single Sign-On (SSO) technologies to authenticate users through custom HTTP headers such as CA SiteMinder. Requires LDAP support.

Standard Atrium Discovery Web Authentication: The user is authenticated by entering a user name and password via the Login page. Supports authentication via LDAP, if LDAP support is enabled.

To configure the web authentication settings:

1. From the Security section of the Administration page, select Single Sign On.
2. Select the Web Authentication tab.

On the Web Authentication page, you can choose the order in which the methods will be attempted, and you can enable, disable, and configure each one. The Standard Atrium Discovery Web Authentication module is a special case (it cannot be disabled and acts as the fail safe login). For each authentication module (except for the Standard Atrium Discovery Web Authentication module), the following controls are provided:

- **Disable** — click this link to disable the module. When a module is disabled, the link is replaced with an Enable link.
- **Configure** — click this link to open a dialog box to configure the module. These dialogs are described in the following sections:
  - SSL Client Certificate Verification
  - SSL Certificate Lookup
  - RSA SecurID Authentication
  - HTTP Header
  - Standard Atrium Discovery Web Authentication
- Ordering controls: click the up or down arrow to move the module up or down. Click the barred up or down arrow to move the module to the top or bottom.

The page also provide links to the configuration pages for HTTPS and LDAP.

### Configuring SSL client certificate verification

This module verifies the client SSL certificate with the web server. If the certificate is valid, the user name is extracted and used for LDAP authorization.
To configure SSL client certificate verification:

1. Click **Configure** in the SSL Client Certificate Verification row.
2. Enter the extract key in the single editable field.
   - The Extract Key which is used to extract the user name. It can be any value in the Distinguished Name (DN) of the supplied X.509 certificate or an X.509 extension value. The default is emailAddress which is used when the email address is the user name.
3. If the user name is not the email address, enter a new extract key to get the user name.
   - This must match the search template used in in the LDAP settings.
4. Click **Apply** to apply the changes.

In 9.0 SP1 and later you can extract values from X.509 certificate extensions. The extension name `subjectAltName` is used as the extract key. The extension name is split into parts. The parts that you can extract are determined by the content of the certificate. For example you can refer to:

- `subjectAltName` — the entire extension name
- `subjectAltName.emailAddress` — email address (as defined in RFC 822 — for example, timothy_taylor@bmc.com "Taylor, Timothy"

⚠️ **Note**

A colon is assumed to delimit fields in the `subjectAltName` value so the string will not be split correctly if a value contains a colon.

### SSL certificate lookup

This module extracts information from the client SSL certificate and verifies it against the LDAP server.

1. Click **Configure** in the SSL Certificate Lookup row.
2. Enter the lookup expression.
   - The lookup expression must be a valid LDAP query. It can contain any values from the supplied X.509 certificate or an X.509 extension value. The variables you can use are:

<table>
<thead>
<tr>
<th>HTTPS</th>
<th>SSL_PROTOCOL</th>
<th>SSL_SESSION_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL_CIPHER</td>
<td>SSL_CIPHER_EXPORT</td>
<td>SSL_CIPHER_USEKEYSIZE</td>
</tr>
<tr>
<td>SSL_CIPHER_ALGKEYSIZE</td>
<td>SSL_VERSION_INTERFACE</td>
<td>SSL_VERSION_LIBRARY</td>
</tr>
<tr>
<td>SSL_CLIENT_M_VERSION</td>
<td>SSL_CLIENT_M_SERIAL</td>
<td>SSL_CLIENT_S_DN</td>
</tr>
<tr>
<td>SSL_CLIENT_S_DN_x509</td>
<td>SSL_CLIENT_I_DN</td>
<td>SSL_CLIENT_I_DN_x509</td>
</tr>
<tr>
<td>SSL_CLIENT_V_START</td>
<td>SSL_CLIENT_V_END</td>
<td>SSL_CLIENT_A_SIG</td>
</tr>
<tr>
<td>SSL_CLIENT_A_KEY</td>
<td>SSL_CLIENT_CERT</td>
<td>SSL_CLIENT_CERT_CHAINn</td>
</tr>
<tr>
<td>SSL_CLIENT_VERIFY</td>
<td>SSL_SERVER_M_VERSION</td>
<td>SSL_SERVER_M_SERIAL</td>
</tr>
</tbody>
</table>
These are the Apache mod_ssl variables. See the Apache website for more information.

3. Enter the LDAP Attribute against which to check the user name.
4. Click Apply to apply the changes.

To configure RSA SecurID authentication

BMC Atrium Discovery can use an RSA SecurID server to perform authentication. To do this you must first install the RSA Authentication Agent 7.1 for Web for Apache Web Server on the appliance, configure it to access your RSA Authentication Manager, and test to ensure that it is working. See the RSA documentation for instructions on how to do this.

Cannot use system and other standard users

You cannot access the system user and the other standard users unless they have an exactly corresponding RSA/LDAP user. You must create an RSA/LDAP user with permissions exactly corresponding to any default users that you use.

To configure RSA SecurID authentication:

1. Log in to the BMC Atrium Discovery UI using an LDAP account with permissions equivalent to the system user. Ensure you can access the Administration -> Web authentication page while logged in as this user.
2. Click Configure in the RSA SecurID Authentication row.
   There is a single editable field in the configure page, this is the Logout URL which is required to logout via the web authentication framework. The default is /webauthentication?logoff?referrer=/ui.
3. Log out of the BMC Atrium Discovery UI.
4. Install and configure the RSA Authentication Manager according to the instructions in the documentation contained in the download.
   - During the configuration of RSA SecurID, "Use RSA Token for Cross-Site Request Forgery Protection" must be set to disabled otherwise logging out from the BMC Atrium discovery UI will fail.
   - The installation requires that some environment variables are configured. These variables should be appended to /etc/sysconfig/httpd. A typical entry looks like this:
# RSA enablement

```bash
# RSA enablement
export VAR_ACE=/var/ace
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/etc/httpd/rsawebagent
```

- If the appliance is a virtual machine and you use VMware snapshots, you should ensure that you update the snapshot after configuring the RSA Authentication Manager. Rolling back to an earlier snapshot removes the shared secret and prevents subsequent log ins. See the RSA Authentication Manager documentation for more information.

5. Navigate to the BMC Atrium Discovery URL. You are presented with the RSA SecurID login page.

6. Log in using the same LDAP account with permissions equivalent to the system user. You are now presented with the standard BMC Atrium Discovery login page.

7. Log in to BMC Atrium Discovery with the same LDAP account as you used in the previous step.

8. Navigate to the Administration -> Web authentication page and enable the RSA SecurID integration.
   
   If you cannot access the Administration -> Web authentication page, you must log out of BMC Atrium Discovery, log back in as the system user, and grant sufficient permissions to the RSA/LDAP user to access that page.

Once RSA SecurID Authentication is enabled in BMC Atrium Discovery, the BMC Atrium Discovery login screen is no longer displayed. To login, enter your username, password, and code from the SecurID token in the RSA SecurID login screens. You are authenticated against the RSA Authentication Manager, and once authenticated you are logged into BMC Atrium Discovery using the same username.

If RSA SecurID Authentication is not enabled, the normal BMC Atrium Discovery login page is displayed, even after successfully logging in using the RSA Authentication Agent. If RSA SecurID Authentication is enabled in ADDM, but the RSA Authentication Agent is not installed or is installed incorrectly, the normal BMC Atrium Discovery login page is also displayed.

### Configuring user authentication using HTTP Header

This section contains instructions on how to integrate BMC Atrium Discovery with single sign on (SSO) technologies which provide authentication using custom HTTP headers such as CA SiteMinder.

The HTTP header plugin scans each HTTP request for a specific HTTP Header. If the HTTP header is present and contains a valid user ID, the user is authenticated; if not, the user is not authenticated. The header is assumed to contain the username or user ID which is used in an LDAP query to obtain authorization. The LDAP query uses LDAP group mapping.

⚠️ **Warning**
HTTP header authentication is a simple authentication mechanism which requires additional protection.

- HTTPS must be enabled with HTTP redirection.
- LDAP support must be enabled
- A reverse proxy must be used, and BMC Atrium Discovery configured only to accept HTTP requests from the IP address or addresses of the proxy. Enabling HTTP header authentication without securing the appliance in this manner leaves the appliance vulnerable to attack.

Example HTTP headers

The SSO application inserts a custom header into each HTTP request. For example:

- Big Corp Inc. uses **BIGUID: 123456**
- Little Corp Inc. uses **LITTLEUSER: fbloggs**

To configure SSO using HTTP header

Before configuring and enabling HTTP header authentication ensure that you understand the potential security implications of this authentication mechanism. To configure HTTP header authentication:

1. On the appliance, click **Administration > Single Sign On**.
2. Click **Web Authentication**.
3. In the HTTP Header row, click **Configure**.
4. Ensure that you understand the potential security implications of this authentication mechanism.
5. In the HTTP Header field, enter the name of the header to use for authentication. This is the header that the SSO application must populate with a valid user ID. BMC Atrium Discovery uses the value of this header to do a lookup in the LDAP server for authentication and for authorization via LDAP group mapping.
6. To complete the configuration, click **Apply**.
7. To enable HTTP header authentication, click **Enable**.

Standard Atrium Discovery web authentication

No configuration is required for the Standard Atrium Discovery Web Authentication section, it is the fail-safe method of logging in to the system. This authentication method uses local users created on the appliance.
Viewing active sessions

Multiple users can be logged in to the BMC Atrium Discovery user interface (UI) concurrently. You can check who is logged in and see details of their session using the Active Sessions page. To view active sessions, click Active Sessions from the Security section of the Administration tab.

The Active Sessions page contains the following information about each user logged in to the UI:

- **User** — The user name of the user logged in.
- **IP Address** — The IP address that the browser is reporting.
- **Browser** — Browser information, including type, major version, platform, language, version and so forth.
- **Login Time** — The time that the user logged in to the UI.
- **Last URL Accessed** — The last URL that the user accessed.
- **Last Access Time** — The time that the user last performed an action in the UI.

Auditing the appliance

Users with sufficient privileges can modify an appliance configuration in ways which could affect the appliance or the customer environment. The appliance audit feature enables you to track changes to the appliance’s configuration. All user-initiated events that modify the state or the behavior of the appliance are logged.

To use the appliance audit feature, you must be logged in as a system user. If you are not a member of this group you are shown the message, You do not have permission to run audit reports.

Reporting on audit events

You can configure the actions that will occur when the appliance status changes. To do this:

From the Security section of the Administration tab, select Audit.

To Search for events, enter search criteria in all or some of the following fields:

- **From** — the start date and time of the search. The default for this field is 24 hours before the page was loaded.
- **To** — the end time and date of the search. The default for this is to display the following text in the To fields: Day Month Year hh mm. This means that the logs will be searched up to the current time.
- **User ID** — a filter to search only for events logged to a particular user, for example, the reporter user.
- **Event group** — a drop-down filter to search only for events belonging to a particular event group or category. The event group provides a means for viewing related event types. See Event Groups for a list of event groups.
- **Events** — a drop-down filter to search only for events of a specified type.
When you have entered the search criteria, click **Run** to start the search. The page is refreshed to show a results table below the search panel.

You can only search the logs through the user interface (UI) using the fields in the Search audit records page. However, if you export the Results List by clicking **Export as CSV**, you can use a spreadsheet or text editor to perform detailed searches on the data. For example, you can search for events on a specific host.

Click **Export as CSV** and choose a location to save the file.

Each item in the result row is a hyperlink to the detailed record of the event.

The record data is divided into two sections:

- **Standard Details**
- **Additional Details**

**Standard details**

The standard details that are recorded for every event are described in the table below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event</td>
<td>The type of event.</td>
</tr>
<tr>
<td>Event Group</td>
<td>The event group to which this event belongs. The purpose of the event group is to provide a filter for viewing related event types.</td>
</tr>
<tr>
<td>User</td>
<td>The user ID who initiated the event.</td>
</tr>
<tr>
<td>Full Name</td>
<td>The full name of the user who initiated the event.</td>
</tr>
<tr>
<td>User Groups</td>
<td>The name of the groups the user who initiated the event belongs.</td>
</tr>
<tr>
<td>When</td>
<td>When the event was logged.</td>
</tr>
<tr>
<td>Summary</td>
<td>Summary description of the event.</td>
</tr>
</tbody>
</table>

**Additional details**

The details shown in the Additional Details section varies from event to event. For example, the following information is provided for a Windows proxy that has been pinged:

- IP Address
- Port
- Windows proxy Name
- Windows proxy Type
When logging in to the user interface over an IPv6 connection, the client might use a temporary IPv6 address. It is this temporary IP address that is reported in the appliance’s audit log. Where temporary addresses are shown, tracing the particular computer from which the login came is difficult. To avoid this, you can disable temporary IPv6 addresses on client computers.

**Event groups**

Audited events are collected into the following groups:

- Appliance Config
- Audit Log
- Consolidation
- DIP
- Datastore Edit
- Discovery Config
- Discovery Ruleset
- ECA Reasoning
- Search
- Security
- Windows proxy
- UI Access
- cmdb-export

The events that belong to these groups are shown on the Audit page in the user interface.

**Purging the audit Log**

You can purge the audit log of all events that are over one month old. Events less than one month old cannot be deleted. You can purge events using the Audit Purge page. To access the Audit Purge page, from the Audit section of the **Administration** tab, select **Purge**.

In the Audit Purge page, the log name, number of events, and the date and time of the oldest record is displayed. A selection drop-down list is displayed which enables you to select the purge until date. The following options are available:

- 1 month ago
- 3 months ago
- 6 months ago
- 12 months ago
- 24 months ago

This ensures that there is a minimum retention period of one month. Click **Purge** to purge the archive up to the Purge until date selected. When you click **Purge**, the operation commences immediately. You can navigate away from the page and continue with other tasks.

Purging archive information is also an auditable event. Therefore, after a purge, the newest event is a record of that purge.
There is no automatic purge of the audit information. When the audit information on the appliance becomes very large, you can use the appliance backup feature to create an archive.

A typical number of auditable events is approximately 1000 per day. This equates to approximately 90 000 events in three months. When deleting events, you can typically remove 500 events per second. Deleting 60 000 or more events will result in the browser timing out, however, the process continues.

Maintaining the appliance

The BMC Atrium Discovery appliance requires little daily maintenance; however, a maintenance mode is provided that only system users can access to perform tasks such as rebooting or shutting down the appliance. In this mode, you can also take appliance backup, and configure the way the system is monitored and audited.

- Setting the appliance identification
- Viewing the appliance specification
- Viewing network interface and routing settings
- Configuring name resolution settings
- Configuring mail settings
- Modifying JVM settings
- Configuring usage data collection
- Localizing the appliance
- Configuring dependency visualizations
- Configuring audit and application options
- Configuring model maintenance settings
- Configuring disk space monitor
- Managing disks and swap space
- Managing disks and swap space in a cluster
- Backing up and restoring the appliance
- Backing up and restoring a cluster
- Shutting down restarting and maintenance mode
- Shutting down restarting and maintenance in clusters
- Performing time synchronization
- Baseline configuration
Setting the appliance identification

Changing the appliance hostname

Change the appliance hostname using the General Settings > Change Hostname options in the Tideway Appliance Network Administration Shell. For more information, see topic on the netadmin user account.

⚠️ The netadmin user is the preferred way of configuring networking

The preferred way of administering any of the appliance network configuration is to use the netadmin user account. The netadmin user account enables you to change network, hostname and gateway settings without requiring root privileges. You can also use the netadmin user to reboot the appliance.

Viewing the appliance identification

You can view or set details of the appliance identity, support information (displayed in the Help drop-down on each BMC Atrium Discovery page), and read-only information about the appliance software and hardware configuration. This includes information about whether the appliance is correctly specified for its intended use.

⚠️ Consolidation and Scanning Appliances

You cannot change the name of an appliance if it is configured as a consolidation or scanning appliance. See Consolidation for more information.

To view or edit the appliance identification:

1. Click Administration.
2. From the Appliance section, click Configuration > Identification.
   The Identification page displays details of the appliance identity, support information, software and hardware.
3. Edit the fields that you want to change.
   The following fields can be edited:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appliance</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>The name of the appliance. You cannot change the appliance name if it is part of a cluster.</td>
</tr>
<tr>
<td>Description</td>
<td>A description of the appliance.</td>
</tr>
<tr>
<td>Field name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Admin Email</td>
<td>An e-mail address for the person or group responsible for the administration of the appliance.</td>
</tr>
<tr>
<td>Banner color</td>
<td>The color for the top banner in the user interface. Setting a banner color makes it easy for users in the field to identify the appliance they are using for various purposes (for example, development, test, and production environments). The appliance name is displayed on the colored banner. See Configuring the banner color for more information.</td>
</tr>
<tr>
<td>Support URL</td>
<td>The URL to use in the help drop down.</td>
</tr>
<tr>
<td>Support URL Title</td>
<td>The title for the support URL.</td>
</tr>
<tr>
<td>Support URL Label</td>
<td>The label for the support URL.</td>
</tr>
<tr>
<td>Support URL Description</td>
<td>The descriptive text label for the support URL.</td>
</tr>
</tbody>
</table>

### Viewing the appliance specification

From the user interface you can view the appliance specification table, which displays the amount of CPU, RAM, SWAP, and DISK resources used by various classes of appliance deployments (such as Proof of Concept, Baseline, Datacentre, and Consolidated Enterprise).

To view the appliance specification:

1. Click **Administration**.
2. In the Appliance section, click **Configuration > Identification**.
   The Appliance Configuration page displays the following sections:
   - Appliance
   - Support
   - Software
   - Hardware

At the bottom of the Hardware section, the application specification table displays the number of discovered Operating System Instances, and the amount of processor, memory, swap space and filesystem resources detected. Also displayed is the threshold value of these parameters for each class of deployment.
This screen illustrates the appliance specification page on a standalone appliance. If the appliance does not have sufficient resources for one or more classes of appliance deployment, the following types of messages are displayed:

- Message at the top of the application specification table: Specifies which resources do not meet the minimum resource specification. For example, if sufficient resource is not available for DISK, the warning message *Below minimum specification (Disk Capacity)* is displayed.
- Warning message at the bottom of the application specification table: Specifies that the appliance has insufficient resources and points to the documentation link for information about hardware specifications.

In the application specification table, the classes of the appliance deployment that do not meet the minimum hardware specification are highlighted in *red*. The classes of appliance deployment that meet or exceed the respective minimum hardware specification are highlighted in *green*.

See the Configuring the Virtual Appliance section for more information about classes of appliance deployment and related topics.

In a cluster, the appliance specification table is not shown.

This screen illustrates the appliance specification page on a cluster member.

**Configuring the banner color**

You can set the color for the top banner in the user interface, making it easy for users in the field to identify, through a visual cue in a prominent banner, the appliance they are using for various purposes (for example, development, test, and production environments).

To configure the banner color:

1. On the Administration tab, click Configuration in the Appliance section.
2. In the Banner Colour section, select the option button corresponding to the color you want to display in the banner and click Apply.
2. In the Name field, type a name that you want to display overlaying the colored banner. The selected banner color with the specified name persists so that all pages that you display in the user interface use the same scheme. The banner color also displays on the login page.

This screen illustrates the selected banner color and name persisting to another selected page in the user interface.

Viewing network interface and routing settings

You can view details of the network interfaces and routing details configured on the appliance.

1. Click Administration.
2. From the Appliance section, click Configuration > Network Interfaces.
   The Network Interfaces page displays details of the configured network interfaces.

Configuring name resolution settings

To configure name resolutions settings:

1. Click Administration.
2. From the Appliance section, click Configuration > Name Resolution.
3. In Search, enter the name of the domain to be searched.
4. In Name Servers, enter the IP address of the name server. Multiple IP addresses can also be specified, separated by a space.
5. From the Time Out drop-down, select the required time out for DNS requests. This can be any value between 1 and 10 seconds. The default is 5 seconds. Larger values can affect Discovery performance and are not recommended.
6. From the Retries drop-down, select the required number of retries after DNS look up failure. This can be any value between 1 and 5. The default is 2. Larger values can affect Discovery performance and are not recommended.
7. Click Apply to save the changes.
Configuring mail settings

To configure mail settings:

1. Click **Administration**.
2. From the Appliance section, click **Configuration > Mail Settings**.
3. Select **Mail Enabled**.
4. In **From Address**, enter the e-mail address used for sending e-mail.
5. In **SMTP Server**, enter the details of the SMTP server.
6. In **SMTP Port**, enter the SMTP port number.
7. If the SMTP server requires authentication, select **Required** in the Authentication section. Then enter the user name and password for the SMTP server.
8. Click **Apply** to save the changes.

⚠️ If you configure the appliance mail server settings for an invalid mail server, using `mail.send()` in a pattern introduces a delay of approximately three minutes while the appliance attempts to contact the SMTP server.

Modifying JVM settings

In some unusual situations, you might encounter errors caused by JVMs running out of memory. You can adjust the amount of memory allocated to individual JVMs up to a limit of 1024MB (the default value depends on the particular JVM). You should increase the JVM memory only if directed by **Customer Support** in response to errors.

1. Click **Administration**.
2. From the Appliance section, click **Configuration > JVM Settings**.
   The JVM Settings page lists the default and current values for the minimum and maximum memory settings for each JVM.
3. To adjust a setting, select the value from the menu in the Change To column.
4. Click **Apply**.
5. Restart the appliance.

Configuring usage data collection

To help BMC Software better understand the ways in which BMC Atrium Discovery is used in customer environments, the Usage Data Collection feature enables submission of anonymous usage data to BMC Software.
Usage data submission

By default, usage data collection is not activated on the appliance. On activation, the appliance submits usage data once every seven days through the customer's web browser. Ten seconds prior to an attempted submission, the Discovery user interface (below the Logout button) displays a message showing the time remaining until the next submission.

To postpone the submission for a day, click not today. Click more info to pause the countdown and show the data to be submitted.

Anonymity

The data sent to BMC Software is totally anonymous. An appliance ID is used, but the appliance cannot be identified from the appliance ID, it is only used to group submissions from the same appliance. The webserver to which the data is sent is configured not to record the source IP address of submitted data.

FAQ

Why?

BMC Atrium Discovery is a high risk system that has access to some very sensitive data in my data center — why does BMC have functionality to share information from BMC Atrium Discovery across the internet?

The critical point to notice here is that BMC Atrium Discovery is not sharing any sensitive data — there are no details in the data about your environment or IT, and there is nothing in the data that can identify its source. Even the web servers at BMC that receive the submissions of the data are configured to discard (that is, to not log) the originating IP addresses, so that we at BMC would be unable to tie a submission to the customer that submitted it even if we wanted to.

The reason that we have this functionality is so that we can see how our customers use the software, and how to optimally make improvements to it — this will be to your direct benefit, as we can ensure we cater to environments of your size.

What is and what can be sent to BMC?

The data that is communicated back to BMC falls into two categories:

- Statistical usage information: this is information such as the version of BMC Atrium Discovery, the number of discovery runs configured in BMC Atrium Discovery, the number of credentials, which reports are used, the number of different types of items discovered (that is, the server count, switch count, and so on) and how many of those are synchronized to a CMDB.
• SNMP Recognition Rules: you can enable or disable this item as required. When you associate a sysOID with a device vendor and model name using the new SNMP recognition rules functionality, that data is sent to BMC for us to include in our content. In this way, other customers will benefit from your capturing this information, and you will benefit from all of the work that they do.

One of the items we send back is the "applianceID" - this is the consolidation ID that is generated by your appliance, and we use it to tell when we have received multiple submissions from the same BMC Atrium Discovery appliance. We cannot use the ID to identify you, as we never receive that ID from you through any other avenue.

Isn't this a half open backdoor that could be exploited further?

Absolutely not. The data to be sent is hard-coded and displayed in its entirety in the UI. The phone-home logic gathers the data it was written to gather, and then it makes an HTTP POST to the BMC webserver to submit it (using a form submission). It does not receive any result from BMC that it then processes, and there is no facility for BMC to instruct the appliance on what data to send.

How can I ensure that this functionality is not switched on by mistake?

We hope that, having read this FAQ, you'll be happy to switch it on deliberately! If you decide not to participate then you can turn the feature off and it will never ask again to be turned on. Further, only somebody with System permissions can enable this feature which protects you from unauthorized staff switching it on.

Managing usage data collection

You can manage data collection from the Administration > Configuration > Usage Data Collection tab. The following table describes the fields and their corresponding functions on the Usage Data Collection tab:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Displays whether usage data collection is activated or not, who made the previous change, and when.</td>
</tr>
<tr>
<td>Submission Data</td>
<td>Displays the data submitted by the appliance.</td>
</tr>
<tr>
<td>Include SNMP Recognition</td>
<td>Select this check box if you want to submit data on SNMP recognition rules.</td>
</tr>
<tr>
<td>Recent Submission History</td>
<td>The previous ten data submission history and links to the corresponding audit report.</td>
</tr>
<tr>
<td>Recent Submission History</td>
<td>Submission Time: The submission time in the Day Month Date Hour Minutes Seconds Year format. To view the audit report of a submission, click the corresponding submission time entry.</td>
</tr>
<tr>
<td></td>
<td>User: The user who was logged in when the data was submitted. To view the audit report of a submission, click the corresponding user entry.</td>
</tr>
<tr>
<td>Activate, Deactivate, and</td>
<td>To activate usage data collection, click Activate.</td>
</tr>
<tr>
<td>Cancel</td>
<td>To deactivate usage data collection, click Deactivate.</td>
</tr>
<tr>
<td></td>
<td>To go back to the main Administration tab, click Cancel.</td>
</tr>
</tbody>
</table>
Activating and deactivating the Usage Data Collection feature

Initial activation
To activate or deactivate usage data collection, you must have permissions to write system settings (system/settings/write). After the initial deployment, when you log on to BMC Atrium Discovery, a Usage Data Collection popup window asks you to activate the feature. Select one of the following options:

- **Activate** — Activates the feature and allows submission of anonymous usage data.
- **Not now** — Postpones the decision to activate the feature until later. The popup window prompt will continue to appear until you activate or deactivate the feature.
- **here** — Displays the Usage Data Collection tab which shows the data that would be submitted on activation.

Subsequent activation and deactivation of the Usage Data Collection
You can choose to deactivate the feature from the Usage Data Collection tab by clicking **Deactivate**. If you want to activate the feature again, click **Activate**.

Configuring anonymous user feedback
The User Feedback feature enables you to submit anonymous comments on BMC Atrium Discovery to BMC. In doing so you can enable us to better understand how BMC Atrium Discovery meets your needs.

User feedback submission
User feedback submission is enabled as a part of usage data collection. By default, the user feedback button is disabled on the appliance. On activation, the user feedback button appears as a floating label in the bottom right of the UI. Any feedback sent this way is entirely anonymous.
Enabling user feedback submission

To enable user feedback submission:

1. You can enable user feedback submission from the Administration > Configuration > Usage Data Collection tab.

2. Usage Data Collection must be activated to enable user feedback submission. If it is not activated, click the Activate button at the bottom of the page.

3. Enable the User feedback capability by selecting the Activate feedback button on all pages option.

Localizing the appliance

This section describes setting localization options such as the keyboard layout and the timezone. The UI however cannot be localized.

Setting the keyboard layout

The console keyboard layout can be temporarily changed using the loadkeys command to test that a keyboard layout works correctly.

To change the keyboard layout to a US layout, enter the following command:

```bash
[root@london01 ~]# loadkeys us
[root@london01 ~]#
```

To change the keyboard layout to a UK layout, enter the following command:

```bash
[root@london01 ~]# loadkeys uk
[root@london01 ~]#
```

After you have determined that the layout works correctly, you should make the change permanent. To do so, change the KEYTABLE, MODEL, and LAYOUT variables in the /etc/sysconfig/keyboard file. For example, to change the keyboard layout to a US layout, use the following:
The keyboard mapping files can be found in /lib/kbd/keymaps/i386/ but usually you can use the 2-letter ISO Country Code. See the ISO website to find the code for the country you require. For example, us (United States), uk (United Kingdom), de (Germany), and no (Norway).

### Setting the system timezone

The system-wide timezone in Linux is defined by `/etc/sysconfig/clock`.

The file `/etc/sysconfig/clock` is used by the system during upgrades to ensure that `/etc/localtime` references the correct information. The `ZONE` value in `/etc/sysconfig/clock` must reference one of the timezone data files in `/usr/share/zoneinfo`. These files contain all the timezone and daylight savings rules for a particular location (for example, `/usr/share/zoneinfo/Europe/London` contains all the data for London). These files are part of the base packages installed by the system (they are from the tzdata package in RHEL and Fedora). The format of the `ZONE` entry in `/etc/sysconfig/clock` should match the required zone, ignoring `/usr/share/zoneinfo`, for example, “Europe/London”.

To set the timezone, as the root user, update the value of `ZONE` in `/etc/sysconfig/clock` and then run the `tzdata-update` command and confirm it is correct with `date`. You must restart the tideway service to bring the timezone change into effect. For example, to set the time to New York time:

```bash
[root@london01 ~]# cp /etc/sysconfig/clock /etc/sysconfig/clock.old
[root@london01 ~]# sed -i -e s/ZONE="[^"]*"/ZONE="US/Eastern"/ /etc/sysconfig/clock
[root@london01 ~]# tzdata-update
[root@london01 ~]# date
Thu Oct 29 11:17:27 EDT 2015
[root@london01 ~]# exit
[tideway@london01 ~]$ sudo /sbin/service tideway restart
```

### Setting the system time

You can set the time using the `date` command. For example, to set the current date to ten past twelve on 4 July 2013, enter the following command:

```bash
[root@london01 ~]$ date -s "12:10:00 20130704"
Thu Jul  4 12:10:00 BST 2013
[root@london01 ~]$ $
```
The format for the date string is **HH:MM:SS YYYYMMDD**.

You can also configure the appliance to synchronize the internal clock to an ntp server. See [Configuring the NTP client at the command line](#) for more information.

---

**Do not change the appliance time on to an earlier setting**

After BMC Atrium Discovery has been running and has created nodes in the datastore, you must not change the time to an earlier setting. The transaction scheme in the datastore is based on timestamps and setting an earlier time makes data appear out of date causing many transactions to fail.

---

### Configuring dependency visualizations

The Dependency Visualization links displayed in the Visualization list in the user interface (UI) are generated from a visualization configuration file. For more information, see the [Viewing dependency visualizations](#) section. You can now add new visualizations to this configuration file, edit existing ones or remove them from the visualization list.

---

### Configuration file

1. **Configuration File Location** - The default configuration file can be found in the following location:

   `/usr/tideway/data/default/graph-definition.txt`

   This default configuration file must not be changed.

2. **Configuration File Customisation** - To customize a configuration file, create the following:

   `/usr/tideway/data/custom/visualizations/graph-definition.txt`

   This overlays the default configuration file.

---

### Structure of the file

The file is divided into three sections by rows of equals signs. The first section defines visualization definitions, the second dependency definitions, and the third tooltip definitions. For an explanation of each of these, see the following sections:

---

### Visualization definitions

The visualization definition section contains a number of definitions for visualizations. Each definition takes the form:

```
visualization_name {tab=tab_name}
viewed_node_type
type_of_dependency dependency_target_node_type
type_of_dependency dependency_target_node_type
```
...  

type_of_dependency dependency_target_node_type

- **visualization_name** — the name of the visualization. This is displayed in the visualizations list when viewing a node of the type defined in `viewed_node_type`.
- **tab_name** — the tab under which to display the visualization. Valid values are:
  - Home
  - Application
  - Infrastructure
  - Discovery
  - Reports
  - Setup
  - If no tab is specified, no tab is selected when the visualization is displayed.
- **viewed_node_type** — the type of node for which this visualization is available.
- **type_of_dependency** - the type of dependency, for example, `depended_upon`, or `dependant`.
- **dependency_target_node_type** - the node type that is the target for the dependency.
  For example, in a dependency visualization describing the relationship between a business application running on a host and the host itself, the target is `Host`.
  For example:

  ```plaintext
  Dependency (tab=Infrastructure)
  SoftwareInstance
  depended_upon SoftwareInstance
  dependant SoftwareInstance
  running_on Host

  BusinessApplicationInstance
  depended_upon BusinessApplicationInstance
  dependant BusinessApplicationInstance
  running_on Host
  ```

This defines a visualization called `Dependency` which will be available on the visualizations list when viewing a `Software Instance` or a `Business Application Instance`. When displayed, the Infrastructure tab will be selected.

When you pick the `Software Instance` visualisation, you see the `Software Instance` node as the starting point of the visualisation, and then all `Software Instance` and `Host` nodes related to that `Software Instance`. Exactly which `Software Instance` and `Host` nodes are shown depends on the definition of the `depended_upon`, `dependant`, and `running_on` dependency definitions, which are defined in the dependency definitions part of this file.
The indentation is important and defines the structure of the visualization. An indentation must be composed of spaces, not tabs, and be a multiple of two spaces.

**Dependency definitions**

The dependency definition (type_of_dependency) defines the route from the viewed_node_type to the dependency_target_node_type. For example, the following visualization definition, from the first section of the file, shows all Hosts that a Software Instance is running on:

```
Dependency
    SoftwareInstance
        running_on Host
```

The running_on Host relationship defines the route from the Software Instance to the Host. One of these definitions must exist in the dependency definitions. For example:

```
SoftwareInstance
    running_on Host
    RunningSoftware:HostedSoftware:Host:Host
```

Here, Software Instance is defined as the source node kind and Host is defined as the destination node kind, and the system will use the traversal RunningSoftware:HostedSoftware:Host:Host to get from any Software Instance nodes currently in the visualization to any related Host nodes.

A dependency definition can have multiple traversals chained one after the other in order to get to a node kind that is distantly related:

```
Host
    connected_to Subnet
        DeviceWithInterface:DeviceInterface:InterfaceOfDevice:NetworkInterface DeviceOnSubnet:
```

The visualization will show only Host and Subnet nodes, but in order to get to Subnet nodes from Host nodes, it will traverse through Network Interface nodes.

You can also define a number of alternative paths by listing them on separate lines:

```
SoftwareInstance
    communication SoftwareInstance
        Server:Communication:Client:SoftwareInstance
        Client:Communication:Server:SoftwareInstance
```
Each path will be tried in turn, and all resulting nodes connected to the source node will be returned.

Relationships can be wildcarded in the same way as in a generic search using ::

```
SoftwareInstance
    composed_of DiscoveredProcess
    InferredElement:Inference::DiscoveredProcess
```

Traversal steps can have an optional '+' after them, for example:

```
SoftwareContainer:SoftwareContainment:ContainedSoftware:SoftwareInstance+
```

This causes the step to be followed repeatedly, adding nodes to the set of nodes found so far, until further traversals add no further nodes. The step is always evaluated at least once. A good use of this function is to follow a relationship that forms an arbitrarily deep hierarchy: for example, the Software Instances making up a BAI. Here it is possible to have a first-order Software Instance with a second-order Software Instance as the container, before reaching the BAI. Although not recommended and not often seen, a third-order Software Instance can be inserted between the second order Software Instance and the BAI, and so on. This function will navigate this structure.

**Attributes**

Attributes can be set on a dependency definition that affect how it is rendered. For example:

```
SoftwareInstance
    communication.server SoftwareInstance (color=0f0,left_arrow)
    Server:Communication:Client:SoftwareInstance
    communication.client SoftwareInstance (color=0f0,right_arrow)
    Client:Communication:Server:SoftwareInstance
```

This displays lines between Software Instance nodes representing client/server communications in green, and draws an arrow pointing at the server.

Attributes appear in brackets after the definition of the dependency type and destination node kind. No spaces are allowed between the brackets, and the attributes are a comma separated list of either flags, or key and value.

**Defined attributes are:**
**color**: sets the color of the line in the graph. It takes a three character argument that is the color as an RGB tuple. Each character is the color of that component as a hexadecimal digit (0-9).

**leftArrow**: sets arrow heads on the line, pointing to the source node.

**rightArrow**: sets arrow heads on the line, pointing to the target node.

Both attributes can be set, in which case both ends of the line have an arrow, and neither attribute can be set, in which case the line is plain.

**leftBox**
- traverse to the destination nodes using the given traversal
- represent the source node as a box
- add the destination nodes in the box

**rightBox**
- traverse to the destination node using the given traversal
- represent the destination node as a box
- add the source node in that box

There must only be a single destination node when using rightBox, because it is impossible to turn several destination nodes into separate boxes and put the source node inside each of them. It should be used when it is known there will always be at most one destination node, for example, the Host Container for a Host.

Boxes only appear at the top level; they cannot be nested. If a later definition conflicts with an earlier, for example, attempting to nest boxes or put a node inside two different boxes, the later definition overrides the earlier. The order depends on how the graph is constructed and is not predictable.

The `leftBox` and `rightBox` attributes override the `color`, `leftArrow`, and `rightArrow` attributes.

**Special traversals**

The `:NetworkConnections:: traversal` which can be used with Hosts and DiscoveredProcesses, connects nodes based on observed communication information. Observed communication information is directly discovered, rather than specific communication relationships built by patterns.

`DiscoveredNetworkConnections` and `ListeningPorts` are used to find this information. These types of traversal are quite slow.
Defining traversals

It is recommended that arrows on edges point towards the depended upon node. For example, the Switch on an edge between Host and Switch, or the Software Instance on an edge between BA1 and Software Instance. This will ensure the layout algorithm lays the graph out properly when in hierarchical layout mode.

It makes sense to name traversals as a verb phrase that can be read in the middle of the start and end node kinds (where end is defined as the node with the arrow head and start is the other one). For example, Software Instance running_on Host. This makes it easy to read off what any given edge in the visualization means.

It is also recommended that when defining a pair of traversals for moving between nodes in either direction (for example, from Host to Software Instance and from Software Instance to Host, both using the "RunningSoftware" relationship), that the traversal is named the same thing. In other words, instead of defining the two traversals like this:

```
Host
  has_running_on_it SoftwareInstance (right_arrow)
  Host:HostedSoftware:RunningSoftware:SoftwareInstance
SoftwareInstance
  is_running_on Host (right_arrow)
  RunningSoftware:HostedSoftware:Host:Host
```

define it like this:

```
Host
  running_on SoftwareInstance (left_arrow)
  Host:HostedSoftware:RunningSoftware:SoftwareInstance
SoftwareInstance
  running_on Host (right_arrow)
  RunningSoftware:HostedSoftware:Host:Host
```

The latter form ensures that the edges all point towards the depended upon node (the Host), and also keeps the name the same. If the name, attributes (color and dashed), and source and destination nodes are the same, but the arrow and source and destination nodes are opposites, the two edges will be merged. Without this, the visualization can become ugly as lots of edges will be double edges.

There is one case where a pair of opposite traversals cannot have the same name: when the source and destination nodes are the same node kind. For example:
Ideally the traversals should be called the same thing, as they are opposites of each other. But because the source and destination nodes are the same, there is no way to distinguish them other than to use a different traversal name. In this case, the visualization might end up with two edges between a pair of Software Instances, one pointing one way and labelled "depended_upon" and the other pointing the other way and labelled "dependant".

It is a convention to use the name "xxxxx.box" for "box" traversals, ie. those that put the source or destination nodes in a box by using left_box or right_box. This is because often the non-box case is also required in some visualizations. With this naming scheme, it is clear which form is being used.

**Tooltip definitions**

By default, the attributes shown as a tooltip when a user hovers over a node are read from the summary list defined in the taxonomy. The tooltip section allows that to be overridden. For example:

```
<table>
<thead>
<tr>
<th>Host</th>
<th>name</th>
<th>os_class</th>
</tr>
</thead>
</table>
```

**Configuring audit and application options**

Certain advanced audit and application options can be configured using the Miscellaneous Settings page.

**To set audit and application options**

1. From the Appliance section of the Administration tab, select Miscellaneous.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow Audit logs to be fully purged</td>
<td>Users can purge the audit log of all events up to one month before the current date. This ensures that there is a minimum retention period of one month. You can configure the appliance to permit purging up to the current date. You can enable or disable full purging with this option. The default is No.</td>
</tr>
</tbody>
</table>
| History Compression Threshold | The historical comparison pages 'compress' the raw history into changes. Entries are grouped by user and a sliding time threshold. This time threshold is necessary to enable you to view a number of separate changes made by Reasoning as a single change. Select from the following values in the list:  
  - 15 seconds  
  - 30 seconds |
The History Compression Threshold can be thought of as a sample rate. Select a value providing sufficient granularity to capture the change that you are looking for. For example, if a node is changing state every 20 minutes, a sample rate below this would be required to capture the changes, in this case, choose 10 minutes.

### Visualization cache

Complex visualizations which need to traverse over many relationships to build the data, such as the Application Dependencies - Software View, can be slow to load. Caching of generated visualizations has been introduced to ensure that this can be minimized.

You should note that cached data is not live, so using this feature will result in visualizations which might not be consistent with the current state of the datastore. This is particularly important when using the forever setting.

Changing either visualization cache settings requires a restart of the tideway service to take effect. Restarting the tideway service clears the visualizations from the cache.

### Configuring model maintenance settings

You can modify settings for maintaining your data model, including aging limits, in the Model Maintenance settings of the user interface.

#### Datastore cache

In previous versions the datastore cache setting required manual configuration. It is now managed automatically.

To configure model maintenance settings:
1. From the Model section of the Administration tab, select Model Maintenance.

The options on the page are described in the following table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directly Discovered Data removal</td>
<td>Specify the age past which Discovery Accesses and all the related DDD nodes are removed (the default is 28 days).</td>
</tr>
<tr>
<td></td>
<td>• Quarterly (90 days)</td>
</tr>
<tr>
<td></td>
<td>• Monthly (28 days)</td>
</tr>
<tr>
<td></td>
<td>• Fortnightly</td>
</tr>
<tr>
<td></td>
<td>• Weekly</td>
</tr>
<tr>
<td></td>
<td>If the setting does not match one of these options the value will be shown as &quot;--custom settings--&quot;.</td>
</tr>
<tr>
<td></td>
<td>In a demonstration appliance only, the default setting for aging is Never, meaning that demonstration data never ages out of the system. This setting is not available for any other type of appliance, and demonstration appliances should never be used for anything other than demonstrations.</td>
</tr>
<tr>
<td></td>
<td>For guidelines, see Modifying DDD, host, and software instance aging limits.</td>
</tr>
<tr>
<td>Time before destroyed nodes are purged</td>
<td>Specify the time elapsed before destroyed nodes are purged from the datastore. The default is one year. To change this, choose one of the following periods from the drop-down list.</td>
</tr>
<tr>
<td></td>
<td>• Never</td>
</tr>
<tr>
<td></td>
<td>• 30 days</td>
</tr>
<tr>
<td></td>
<td>• 90 days</td>
</tr>
<tr>
<td></td>
<td>• 180 days</td>
</tr>
<tr>
<td></td>
<td>• One year</td>
</tr>
<tr>
<td></td>
<td>• Two years</td>
</tr>
<tr>
<td>Time before history entries are purged</td>
<td>Specify the time elapsed before history entries are purged from the datastore. The default is never. To change this, choose one of the following periods from the drop-down list.</td>
</tr>
<tr>
<td></td>
<td>• Never</td>
</tr>
<tr>
<td></td>
<td>• 30 days</td>
</tr>
<tr>
<td></td>
<td>• 90 days</td>
</tr>
<tr>
<td></td>
<td>• 180 days</td>
</tr>
<tr>
<td></td>
<td>• One year</td>
</tr>
<tr>
<td></td>
<td>• Two years</td>
</tr>
<tr>
<td>Scan optimization timeout</td>
<td>When an IP address is designated the preferred IP address for a host, Discovery does not perform scans on other known IP addresses for that host. This is referred to as scan optimization. After a specified Scan optimization timeout period, Discovery will scan other known IP addresses for that host to ensure that they are still non-preferred and on the same host. The default is seven days. Specify one of the following timeout periods from the drop-down list.</td>
</tr>
<tr>
<td></td>
<td>• 1 - 10 days</td>
</tr>
<tr>
<td></td>
<td>• 14 days</td>
</tr>
<tr>
<td></td>
<td>• 21 days</td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Specify the aging time limit for host, network device, storage device, printer, SNMP managed device, mainframe, and management controller nodes. This value must be higher than the Scan optimization timeout. The default is 10 days. | • 2 - 10 days  
• 14 days  
• 21 days  
• 28 days  
For guidelines, see Modifying DDD, host, and software instance aging limits. |
| Device aging access failures                   | Specify the access failure limit for host, network device, storage device, printer, SNMP managed device, mainframe, and management controller nodes. The default is seven failures.                                 |
| Time to elapse before a software instance/runtime environment/storage can be aged          | Specify the time to elapse before a software instance/runtime environment/storage node can be aged. (the default is 10 days).                                                                            |
| Minimum number of failed accesses before a software instance/runtime environment/storage is aged | Failed accesses before a software instance/runtime environment/storage node is aged.                                                                                                                     |
| Size Warning                                   | The maximum size of the datastore. When this limit is reached, a warning is displayed in the Appliance Status page. This is a soft limit and does not prevent the datastore growing beyond the specified limit. You can select any value from 10 to 100 GB in steps of 5 GB from the list. The default is 30 GB. |
| Checkpoint Interval                            | The interval between datastore checkpoints. The larger the interval, the more data has to be written to the checkpoint files. The smaller the interval, less data needs to be written to the checkpoint files, though the writing takes place more often. The default is 15 minutes. You must contact Customer Support before you make any changes to this setting. |

**Modifying DDD, host, and software instance aging limits**

In general, the expectation that BMC Atrium Discovery uses for deriving the default model maintenance settings is based on performing one scan of the estate every 24 hours with a DDD depth of one month. This expectation is also used to derive the sizing data. The default setting for data aging for both hosts and software instances is 10 days, because for most deployments this limit provides the best balance of responsiveness without data thrashing. Putting this in a business example, this gives two weekends plus additional time to detect that a host is aging, investigate why it is doing so, and make changes before the host is destroyed. It gives the software teams the same length of time to sort out any failures in the estate.

**Best Practice**
BMC recommends that you do not attempt to fine-tune model maintenance parameters. Doing so can make BMC Atrium Discovery highly reactive and have negative consequences, such as causing a dramatic increase in the size of the datastore and putting more load on your target estate. If you do alter the model maintenance parameters, BMC recommends that you do not vary more than half or twice the standard settings detailed in the preceding table.

However, there might be occasions when modifying the defaults are necessary, particularly if you scan your estate at different intervals and need to keep close control on disk consumption. The following table shows recommended settings based on your scanning frequency.

<table>
<thead>
<tr>
<th>Scanning Frequency</th>
<th>Recommended Aging Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every 2 days</td>
<td>Maintain DDD at 28 days, and decrease the Host and Software Instance access failures to 4 (to account for half as many expected scans).</td>
</tr>
<tr>
<td>2 times per day</td>
<td>Decrease DDD to 14 days (to cap how much space it consumes, unless you increase the disk space to more than the sizing guidelines); increase the Host and Software Instance access failures to 12-14 (to account for scanning at twice the expected rate).</td>
</tr>
</tbody>
</table>

For more information about aging and how it fits in the node removal process, see How nodes are removed.

Scheduled DDD aging

In BMC Atrium Discovery systems where discovery (or consolidation) is in progress for most of the available time, contention between the removal (aging out) of DDD nodes (in this case Discovery Access nodes and their children) and the creation of new nodes, might affect the performance of in-progress discovery runs. To avoid this performance impact you can schedule DDD removal blackout windows during which no DDD removal is undertaken.

When should DDD removal blackout windows be used?

DDD removal blackout windows should typically only be used on consolidation appliances when you need to achieve maximum discovery throughput. In virtually every discovery schedule used, the continual aging scheme used by BMC Atrium Discovery can remove "old" DDD at a similar rate as they are created.

You can determine whether scheduling DDD removal blackout windows might be beneficial using the DDD removal statistics page. The DDD removal statistics page shows the total number of DiscoveryAccesses in the datastore and those eligible for removal. If DiscoveryAccess removal is keeping up with DiscoveryAccess creation, then the number of eligible Discovery Access nodes is zero, or near zero, DDD removal blackout windows are not required. If the trend of eligible DiscoveryAccess is rising over a two week period, then DDD removal blackout windows might be a solution. See DDD removal statistics page for more information.
Viewing existing DDD aging blackout windows

To view existing DDD aging blackout windows:

1. From the Model section of the Administration tab, select Model Maintenance.
2. Select the DDD Removal Blackout Windows tab.

When you view the DDD Removal Blackout Windows tab, the active blackout window is highlighted. If multiple blackout windows are active, the one with the longest time remaining before it ends is highlighted. This is not automatically refreshed.

DDD nodes are removed in batches which are not interrupted. Once removal starts, it continues to completion. Therefore, if a batch removal is in progress at the beginning of a DDD removal blackout window, it will continue into the blackout until completion. In normal operation this should take no more than a few minutes.

Adding a new DDD aging blackout window

You can schedule a new DDD aging blackout window to occur daily, weekly, or monthly and can specify a start time and duration. To schedule a new DDD aging blackout window:

1. Click the Add button. The Add a New DDD Removal Blackout Window dialog box is displayed.
2. Enter the information for the blackout window in the fields described below.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment</td>
<td>Enter a descriptive comment for the blackout window.</td>
</tr>
<tr>
<td>Frequency</td>
<td>Select a frequency for the window to operate. For example, this can be Daily, Weekly, or Monthly. For a weekly blackout window, you are provided with buttons for each day. Select the day or days that you want the window to operate. For a monthly blackout window, you are provided with buttons for each day in the month. Select the day or days that you want the window to operate. Alternatively, select the No Removal On The radio button and choose one of: First, Second, Third, Fourth, Last and the day that you want the window to operate. In this way you can select the Second Tuesday of the month and so forth.</td>
</tr>
<tr>
<td>Start Time</td>
<td>Select a time for the window to start.</td>
</tr>
<tr>
<td>Duration</td>
<td>Select a duration in hours. This is the length of time that the blackout window operates and can be from 1 to 24 hours. For a daily blackout window the maximum number of hours you can select is 23. This prevents you from inadvertently scheduling a 24/7 blackout.</td>
</tr>
</tbody>
</table>
3. Click OK.
Scheduling DDD removal blackout windows

In BMC Atrium Discovery systems where discovery (or consolidation) is in progress for most of the available time, contention between the removal (aging out) of DDD nodes (in this case Discovery Access nodes and their children) and the creation of new nodes, might affect the performance of in-progress discovery runs. To avoid this performance impact you can schedule DDD removal blackout windows during which no DDD removal is undertaken.

When should DDD removal blackout windows be used?

DDD removal blackout windows should typically only be used on consolidation appliances when you need to achieve maximum discovery throughput. In virtually every discovery schedule used, the continual aging scheme used by BMC Atrium Discovery can remove "old" DDD at a similar rate as they are created.

You can determine whether scheduling DDD removal blackout windows might be beneficial using the DDD removal statistics page. The DDD removal statistics page shows the total number of DiscoveryAccesses in the datastore and those eligible for removal. If DiscoveryAccess removal is keeping up with DiscoveryAccess creation, then the number of eligible Discovery Access nodes is zero, or near zero, DDD removal blackout windows are not required. If the trend of eligible DiscoveryAccess is rising over a two week period, then DDD removal blackout windows might be a solution. See DDD removal statistics page for more information.

Viewing existing DDD aging blackout windows

To view existing DDD aging blackout windows:

1. From the Model section of the Administration tab, select Model Maintenance.
2. Select the DDD Removal Blackout Windows tab.

When you view the DDD Removal Blackout Windows tab, the active blackout window is highlighted. If multiple blackout windows are active, the one with the longest time remaining before it ends is highlighted. This is not automatically refreshed.

DDD nodes are removed in batches which are not interrupted. Once removal starts, it continues to completion. Therefore, if a batch removal is in progress at the beginning of a DDD removal blackout window, it will continue into the blackout until completion. In normal operation this should take no more than a few minutes.

Adding a new DDD aging blackout window

You can schedule a new DDD aging blackout window to occur daily, weekly, or monthly and can specify a start time and duration. To schedule a new DDD aging blackout window:
1. Click the Add button. The **Add a New DDD Removal Blackout Window** dialog box is displayed.

2. Enter the information for the blackout window in the fields described below.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comment</td>
<td>Enter a descriptive comment for the blackout window.</td>
</tr>
<tr>
<td>Frequency</td>
<td>Select a frequency for the window to operate. For example, this can be Daily, Weekly, or Monthly. For a weekly blackout window, you are provided with buttons for each day. Select the day or days that you want the window to operate. For a monthly blackout window, you are provided with buttons for each day in the month. Select the day or days that you want the window to operate. Alternatively, select the <strong>No Removal On The</strong> radio button and choose one of: • First • Second • Third • Fourth • Last and the day that you want the window to operate. In this way you can select the Second Tuesday of the month and so forth.</td>
</tr>
<tr>
<td>Start Time</td>
<td>Select a time for the window to start.</td>
</tr>
<tr>
<td>Duration</td>
<td>Select a duration in hours. This is the length of time that the blackout window operates and can be from 1 to 24 hours. For a daily blackout window the maximum number of hours you can select is 23. This prevents you from inadvertently scheduling a 24/7 blackout.</td>
</tr>
</tbody>
</table>

3. Click **OK**.

**Configuring disk space monitor**

To prevent problems that can occur if the disk partition containing the datastore files or log files runs out of space, the free disk space monitor is used. The monitor service uses two predefined thresholds for disk space:

- **Free disk space baseline threshold**: The threshold below which the machine baseline raises a critical warning. By default, it is 2048 MB for the partition holding the datastore and 200MB for the partition holding the log files. Baseline threshold is twice of the free disk space threshold. The baseline threshold only raises a warning and does not shut down the machine or prevent it from starting.
- **Free disk space shut down threshold**: The threshold below which the free disk space monitor gracefully shuts down BMC Atrium Discovery and prevents it from starting until the free disk space baseline threshold is available. By default, it is 1024 MB for the partition holding the datastore and 100Mb for the partition holding the log files.

⚠️ If free disk space on respective partitions on a machine in a cluster is below the shut down threshold, the free disk space monitor shuts down all the machines on that cluster and prevents the machines from starting until the free disk space at least match the baseline threshold.
Managing free disk space

You can perform the following operations related to managing and monitoring the disk space on a standalone machine or a cluster member:

- view the disk usage of a standalone machine
- view the disk usage of the cluster member
- check the Appliance Disk Space of a standalone machine or a cluster member against the baseline
- free up disk space on the machine
- restart the services to recover from the forced shutdown

Viewing the disk usage of a standalone machine

To view disk usage on a standalone machine:

1. Click Administration.
2. From the Appliance section, click Disk Configuration.
   The Current Usage section of the Manage ADDM Disks page displays the current disk usage. For more information, see Managing disks and swap space.

Viewing the disk usage of the cluster member

To view disk usage on a cluster:

1. On the coordinator, click Administration.
2. From the Appliance section, click Cluster Management.
   The Cluster Management page displays the disk usage of each machine on the cluster.

Checking the Appliance Disk Space against the baseline

To check the Appliance Disk Space against the predefined baseline on a standalone machine or any cluster member:

1. Click Administration.
2. From the Appliance section, click **Baseline Status**.
   The Baseline Status page displays the results of the recent checks.

3. Click **Check Baseline Now** at the bottom of the list to get the latest information and see if the Appliance Disk Space check status is **OK**.

**Freeing up disk space**

If the free disk space is below the baseline threshold, a critical baseline warning is raised:

![Baseline Status Page](image)

To prevent any shut down, you must free up some disk space on the machine. Possible areas from where you may free up disk space are:

- **Application log files**: see **Contents of the logs** for a description and the location of the main application log files. BMC Atrium Discovery is not reliant on these log files and they can be safely deleted. However, the datastore transaction logs must not be deleted, see **operational warnings** for more information.
- **Cores**: check that there are no old cores in `$TIDEWAY/cores`.
- **Record data**: you can delete any record data that is not required, or using the **disk configuration tool** you can move record data to an additional disk.
- **Workspaces**: any areas that you have used as workspaces or staging areas for exports.

If the free disk space is below the shut down threshold, the free space monitor shuts the machine down and the UI displays the machine name which has insufficient free disk space. You must log on to that machine and free up the disk space to the baseline threshold at the minimum. Thereafter, you must restart the services.

**Restarting the services to recover from the forced shutdown**

When the free disk space goes below the shutdown threshold which is equal to baseline threshold multiplied by two, the BMC Atrium Discovery services are stopped. After you free up the space on the machine that caused the shutdown, you need to restart these services.

To restart services, run the following command on a standalone machine or on any machine in the cluster:

```
tw_cluster_control --cluster-start-services
```

**Managing disks and swap space**

The disk configuration utility enables you to move BMC Atrium Discovery data onto and create swap space on new disks on your appliance. However, the following restrictions apply:
• You can only move data to or create swap space on disks that are not currently used by BMC Atrium Discovery.
• You cannot move more than one component of BMC Atrium Discovery data to a given target disk.
• You cannot move the components back to the system disk after they have been moved to a secondary disk.

**Back up your data before changing disk configurations**

The disk management utility performs disk operations such as partitioning and moving data. All data on the new disks is deleted. When data is moved from the system disk, it is copied to the new disk and deleted from the system disk. Before using it you should always back up your appliance.

**LVM is not supported**

If you try to add an LVM disk, the operation fails and the following message is displayed:

*Unexpected system configuration: LVM-based configuration detected*

The following disk types are defined:

<table>
<thead>
<tr>
<th>Disk type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Disk</td>
<td>The disk on which BMC Atrium Discovery is installed.</td>
</tr>
<tr>
<td>Additional Disk</td>
<td>Any additional disk which contains BMC Atrium Discovery data.</td>
</tr>
<tr>
<td>New Disk</td>
<td>Any additional disk with no mounted partitions. There must be 4000 MB free</td>
</tr>
<tr>
<td></td>
<td>space on the disk after the copy operation or the operation is not permitted.</td>
</tr>
<tr>
<td></td>
<td>If you are only creating swap space, the disk must be larger than the</td>
</tr>
<tr>
<td></td>
<td>suggested swap size.</td>
</tr>
<tr>
<td>Non ADDM Disk</td>
<td>Any additional disk with mounted partitions. These partitions must be</td>
</tr>
<tr>
<td></td>
<td>unmounted before this disk can be configured using the disk management</td>
</tr>
<tr>
<td></td>
<td>utility.</td>
</tr>
</tbody>
</table>

The Disk Configuration page of a standalone appliance displays the following:

• A panel which shows any pending operations
• A panel which shows the current configuration of any disks in the appliance and the suggested configuration of any new disks
• A comprehensive key to the data types and their representation
On any new disk, a Change Usage drop down list is displayed which is populated with the data partitions on the system disk. Where there is insufficient space on the new disk to copy the data from a particular partition, that partition's label is disabled. The label is also disabled if the partition has already been selected to have its data moved to another disk.

You can also choose to add a swap partition to the disk by selecting the check box. The check box is only displayed if you have insufficient swap space configured. The disk configuration utility uses the following calculation to determine the best swap size.

- Where the amount of memory is less than 16 GB, swap size is set at double the memory size.
- Where the amount of memory is between 16 and 32 GB, swap size is set at 32 GB.
- Where the amount of memory exceeds 32 GB, swap size is set to equal the memory.

The add swap check box is not displayed if the available new disk does not have enough space to store the additional swap to reach the optimal swap size. For example, if the system has 8GB swap by default, and the optimal swap size is 24GB, then 16GB additional swap is required. If the new disk has only 12GB available, the check box is not displayed.

When you make the changes you have chosen or to accept the changes suggested by the system, the services are shut down and a progress indicator is displayed.

To move data to a new disk

1. Backup your appliance.
2. From the Appliance section of the Administration tab, click Disk Configuration. The Disk Configuration page is displayed.
3. If the suggested configuration is what you want, continue this procedure from here.
4. If you do not want to accept the suggested configuration, select Do not move any data here from Change Usage.
5. Select the data to move to the new disk from Change Usage. If the disk is of the type “Non ADDM Disk” then you must unmount all partitions on the disk before proceeding.
6. Select the Add swap check box if you want to add additional swap space.
7. Review the pending changes list.
   If it is what you want, click Start.
8. You are requested for confirmation. The BMC Atrium Discovery Services are stopped and all of the contents of the new disks are deleted before moving data or creating swap partitions. Click **OK** to proceed.

9. The status page is displayed which shows a progress indicator and details of the individual operations. Once the operation has been completed, the services are started and the login page is displayed.

⚠️ If the only operation you perform is to add swap to a new disk, the services are not stopped and started and you do not need to login.

When data is moved from an additional disk that contains only a data partition, at the end of processing, the disk is identified as a "New Disk" and is displayed with a suggested configuration. The existing data is not deleted, though the partition is not mounted.

When data is moved from an additional disk that contains a data partition and a swap partition, at the end of processing, the disk is identified as an "Non ADDM Disk". The existing data is not deleted, though the partition is not mounted. To reuse this disk, disable the swap partition as shown here.

### Managing disks and swap space in a cluster

The disk configuration utility enables you to add disks, move BMC Atrium Discovery data onto new disks and create swap space on new disks on all of the machines in a cluster. All of the operations apply to the individual machine, you cannot move data around the cluster using this utility. However, the changes you make to the individual machines are coordinated, as when you make changes to disks on a cluster member, all members of the cluster need to be shut down.

To minimize cluster downtime, you configure and queue the changes on individual machines and then apply the changes. When you apply the changes, the cluster is shut down automatically and restarted when all of the changes have been applied to the individual machines.

As with a standalone appliance, the following restrictions apply:

- You can only move data or create swap space on disks that are not currently used by BMC Atrium Discovery.
- You cannot move more than one component of BMC Atrium Discovery data to a given target disk.
- You cannot move the components back to the system disk after they have been moved to a secondary disk.

⚠️ **Back up your data before changing disk configurations**
The disk management utility performs disk operations such as partitioning and moving data. All data on the new disks is deleted. When data is moved from the system disk, it is copied to the new disk and deleted from the system disk. Before using it you should always backup your cluster.

- LVM is not supported

If you try to add an LVM disk, the operation fails and the following message is displayed:

```
Unexpected system configuration: LVM-based configuration detected
```

The following disk types are defined:

<table>
<thead>
<tr>
<th>Disk type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Disk</td>
<td>The disk on which BMC Atrium Discovery is installed.</td>
</tr>
<tr>
<td>Additional Disk</td>
<td>Any additional disk which contains BMC Atrium Discovery data.</td>
</tr>
<tr>
<td>New Disk</td>
<td>Any additional disk with no mounted partitions. There must be 4000 MB free space on the disk after the copy operation or the operation is not permitted. If you are only creating swap space, the disk must be larger than the suggested swap size.</td>
</tr>
<tr>
<td>Non ADDM Disk</td>
<td>Any additional disk with mounted partitions. These partitions must be unmounted before this disk can be configured using the disk management utility.</td>
</tr>
</tbody>
</table>

The Disk Configuration page of a cluster member displays the following:

- A panel which shows the queued changes for all members of the cluster
- A drop down list with links to the UI of each of the other cluster members
- A panel which shows the current configuration of any disks in the appliance and the suggested configuration of any new disks
- A comprehensive key to the data types and their representation
Shows the disk configuration utility page of a cluster member.

Configuration options for a new disk

Once a physical (or virtual in the case of a VM) disk has been added to the appliance and recognized by the operating system, it is displayed in the Disk Configuration window.

On any new disk, a Change Usage drop down list is displayed which is populated with the data partitions on the system disk. Where there is insufficient space on the new disk to copy the data from a particular partition, that partition's label is disabled. The label is also disabled if the partition has already been selected to have its data moved to another disk.

You can also choose to add a swap partition to the disk by selecting the check box. The check box is only displayed if you have insufficient swap space configured. The disk configuration utility uses the following calculation to determine the best swap size.

- Where the amount of memory is less than 16 GB, swap size is set at double the memory.
- Where the amount of memory is between 16 and 32 GB, swap is set at 32 GB.
- Where the amount of memory exceeds 32 GB, swap is set to equal the memory.

The add swap check box is not displayed if the available new disk does not have enough space to store the additional swap to reach the optimal swap size. For example, if the system has 8GB swap by default, and the optimal swap size is 24GB, then 16GB additional swap is required. If the new disk has only 12GB available, the check box is not displayed.

When you make the changes you have chosen or to accept the changes suggested by the system, the services are shut down and a progress indicator is displayed.

To move data to new disks on cluster members

Once new disks have been added to cluster members and recognized by their individual operating systems, they are displayed in the Disk Configuration window of that machine's UI. To move data to the disk:

1. Backup your cluster.
2. From the Appliance section of the Administration tab, click Disk Configuration. The Disk Configuration page is displayed.
3. If the suggested configuration is what you want, continue this procedure from here.
4. If you do not want to accept the suggested configuration, select Do not move any data here from Change Usage.
5. Select the data to move to the new disk from Change Usage. If the disk is of the type "Non ADDM Disk" then you must unmount all partitions on the disk before proceeding.
6. Select the Add swap check box if you want to add additional swap space.
7. Click Queue to add the changes to the Queued Changes for the Cluster list.
8. Repeat from here for each member of the cluster that you need to make changes to. To access the other members of the cluster, select the link for the required member from the Go to another Cluster Member drop down list.
9. Review the Queued Changes for the Cluster list. Click one of the following options to continue:
   - Click Cancel all to remove all queued changes for all machines from the list. You can then start the procedure again.
   - Click Cancel to delete the queued request for the current machine and enable you to reconfigure it.
   - Click Start to proceed.
10. If you clicked Start, you are then requested for confirmation. The BMC Atrium Discovery services are stopped on all members of the cluster and all of the contents of the new disks on all members are deleted before moving data or creating swap partitions.
11. The status page is displayed on each cluster member UI which show a progress indicator and details of the individual operations. Once the operation has been completed, the services are started on all members and the login page is displayed on each member's UI.

⚠️ Note

If the only operation you perform is to add swap to a new disk, the services are not stopped and started and you do not need to login, nor do you need to shut down the cluster.

When data is moved from an additional disk that contains only a data partition, at the end of processing, the disk is identified as a "New Disk" and is displayed with a suggested configuration. The existing data is not deleted, though the partition is not mounted.

When data is moved from an additional disk that contains a data partition and a swap partition, at the end of processing, the disk is identified as an "Non ADDM Disk". The existing data is not deleted, though the partition is not mounted. To reuse this disk, disable the swap partition as shown here.
Backing up and restoring the appliance

The appliance backup feature enables you to backup the appliance. You can restore the backed up data when required. The backup can be made to the local file system, or to a remote destination over ssh or via a Windows share, so little local disk space is required. You can choose to be notified by email, if configured, of the completion and result (success or failure) of a backup or restore task.

- A backup can only be restored on the same BMC Atrium Discovery major/minor version and service pack. The patch version of the target appliance can be newer.
- The appliance is shutdown for the backup or restore and restarted when the backup or restore is complete.
- The UI lists the items which will be backed up.

**Destination system time must not be earlier than source when restoring a backup**

You must ensure that the current system time on the destination appliance is no earlier than that of the appliance on which the backup was created. If the modification times on the files contained in the backup are later than the system time when they are restored, the backup will hang. To recover at this point you must kill the backup process, correct the time, run `tw_restore --fix-interrupted` then repeat the restore using the `tw_restore` command line utility.

**Backing up and CMDB synchronization**

The backup contains the CMDB synchronization configuration. When a host has significantly changed so that its key has also changed, problems can be caused if a backup is restored before the changed host is rediscovered. In this case, on the next CMDB synchronization, duplicate hosts will be created in the CMDB representing the changed host, and the CIs representing the original hosts will never be deleted. To ensure that no duplicate hosts are created, you can delete and then recreate the BMC ADDM dataset.

The backup contains the LDAP configuration. If the destination appliance cannot access the LDAP server, you must ensure that a local (non-LDAP) user belonging to the system and public groups is activated and successfully tested on the source appliance before making a backup.

If you choose to exclude sensitive data when backing up an appliance in which CMDB
synchronization has been configured, the CMDB Sync page on the restored appliance displays the "This appliance has not been set up for synchronization with the Atrium CMDB" message. Once the Setup form is complete, filter and blackout window settings are restored.

The appliance backup feature replaces the appliance snapshot that was available in previous releases. On upgraded appliances only, if snapshots are still held in the filesystem, a banner and Remove Snapshots button is provided so you can remove the snapshots and release disk space.

To create a backup of the appliance

1. From the Appliance section of the Administration tab, click Backup & Restore. The Appliance Backup page is displayed. This has a panel in which you can configure the backup destination, and details of the size and contents of the backup.

   Shows the create backup tab of the appliance backup page.

2. Enter the details for the backup destination. The fields that can be completed are displayed or hidden depending on the backup type selection. Required fields are indicated with a red asterisk.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
   | Backup Type | Select the destination type from the drop down list. This can be one of the following:  
   | [ ] Local — The backup is written to the $TIDEWAY/var/backup directory. Only one local backup can be stored.  
   | [ ] SSH — The backup is written to a remote server over ssh.  
   | [ ] Windows Share — The backup is written to a Windows share.  
   | Notes | A free text area in which you can write notes about the backup.  
   | Directory | The directory into which to write the backup on the remote SSH server. Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup inside the specified directory. (SSH only).  
   | Path | The share name and directory name into which to write the backup on a Windows share. Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup inside the specified directory. Path syntax is \sharename\directoryname, where sharename is the name of the Windows share, and directoryname is the name of the directory into which to write the backup. (Windows share only).  
   | Host | The hostname or IP address of the remote server onto which to write the backup (SSH).
### Field Name | Details
--- | ---
Port | The port to which to connect (SSH).
Username | The username to use to connect to the remote server (SSH and Windows share). To specify the domain for Windows shares, use the following syntax: **user@domain**
Password | The corresponding password (SSH and Windows share).
Verify backup | Select **Verify backup** to verify (md5) that the files in the backup archive are the same as those on the appliance. Shown only after successfully testing the connection for SSH and Windows share backups.
Exclude sensitive data | Select **Exclude sensitive data** to exclude sensitive data from the backup. This will exclude the vault and the appliance key and certificate. Appliance UI users are always backed up and restored, regardless of this setting. So, for example, after a restore the password in effect for the system user will be the one from the source appliance. Shown only after successfully testing the connection for SSH and Windows share backups.
Email when complete | Select **Email when complete** and enter an email address if you want an email to be sent automatically when the backup task is completed. Shown only after successfully testing the connection for SSH and Windows share backups.
Test Connection | Click **Test Connection** to ensure that the remote host can be contacted and that the credentials are valid. Shown only for SSH and Windows share backups.

3. Click **Shutdown & Backup** to start the backup operation.
   You are prompted for confirmation.
4. Click **No** to return to the Appliance Backup page. Click **Yes** to continue and backup the appliance.
   All services are shut down before the backup occurs and a progress screen is displayed. A **Cancel** button is also displayed, but is only enabled at the stages of the backup where it is possible to cancel.

### To restore a backup to the appliance

⚠️ **FIPS enabled appliance**

You should not attempt to restore a backup to a FIPS enabled appliance. Rather, you should restore a backup onto the appliance and then **enable FIPS**. If you restore to a FIPS enabled appliance, the services cannot restart and the progress screen shows an `omniORB.CORBA.INITIALIZE` error.

1. Make sure that the time setting on the destination appliance is not earlier than the source appliance. Failing to do so results in a failed restore, and a time consuming process to repair the restore. See this warning for more information.
2. From the Appliance section of the **Administration** tab, click **Backup & Restore**.
   The Appliance Backup page is displayed.
3. Click the **Restore Backup** tab.

The **Restore Backup** tab has a panel in which you can choose the source of the backup, and provides details of the size and contents of the existing local backup.

![Restore Backup Tab](image)

*Shows the restore backup tab of the appliance backup page.*

4. Enter the details for the backup source. The fields that can be completed are displayed or hidden depending on the backup source type selection. Required fields are indicated with a red asterisk.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
| **Backup Type**  | Select the destination type from the drop down list. This can be one of the following:  
• **Local** — The backup is read from the $TIDEWAY/var/backup directory.  
• **SSH** — The backup is read from a remote server over ssh.  
• **Windows Share** — The backup is read from a Windows share. |                                                                                                                                               |
| **Directory**    | The directory from which to read the backup on the remote server (SSH). Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup inside the specified directory. Ensure you specify the subdirectory name too. (SSH only). |                                                                                                                                               |
| **Path**         | The share name and directory name from which to read the backup on the remote server (Windows share). Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup inside the specified directory. Path syntax is \sharename\directoryname, where sharename is the name of the Windows share, and directoryname is the name of the directory containing the backup. (Windows share only). |                                                                                                                                               |
| **Host**         | The hostname of the remote server from which to read the backup (SSH).                                                                                                                                 |
| **Port**         | The port to which to connect (SSH).                                                                                                                                                                        |
| **Username**     | The username to use to connect to the remote server (SSH and Windows share). To specify the domain for Windows shares, use the following syntax: user@domain                                                                 |
| **Password**     | The corresponding password (SSH and Windows share).                                                                                                                                                        |
| **Preserve Identity** | Select the check box to preserve the appliance's identity rather than take on the identity from the restored backup. This consists of:  
• Appliance identity  
• HTTPS configuration  
• Consolidation configuration  
Clear the check box to use the identity from the backup. |                                                                                                                                               |
| **Email when complete** | Select the check box and enter an email address if you want an email to be sent automatically when restoring the backup is complete. |                                                                                                                                               |
4. **Details**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Connection</td>
<td>When you enter valid connection information, the Test Connection button is enabled. Click this to test the connection to the remote backup server. When the test is successful, and a backup is present, the Remote Backup Details pane displays information on the remote backup.</td>
</tr>
</tbody>
</table>

5. **Click Shutdown & Restore** to start the restore operation. You are prompted for confirmation.

6. **Click No** to return to the Appliance Backup page. **Click Yes** to continue and restore the appliance.

   All services are shut down before the restore occurs and a progress screen is displayed.

---

### Device package error on restore

Network device definitions are automatically installed with a TKU. The source and destination appliance must have the same version network device definitions. If you see the following error when restoring, it is important to update the package as described in the error message.

```
An error was reported while installing the device package.
Please correct the error and install the device package manually by running (as root):
   tw_device_import --force -U --oldpackage <rpm file>
```

The device package is located in the \$TIDEWAY/data/installed/tpl directory.

---

### Backing up and restoring a cluster

The appliance backup feature enables you to backup the appliance. You can restore the backed up data when required. The backup can be made to the local file system, or to a remote destination over ssh or via a Windows share, so little local disk space is required. You can choose to be notified by email, if configured, of the completion and result (success or failure) of a backup task.

If you use appliance backup on a cluster, the entire cluster is backed up. If you choose to backup to local filesystem (the On Member option), each cluster member is backed up onto its local filesystem. When you backup a cluster to a remote destination, the appliance backup feature creates a single backup in which each machine’s backup is contained in a subdirectory.

You can cancel an in-progress cluster backup, but only from original session on the UI of the machine from which the backup was started.

---

### VMware snapshots and clusters


You can use the VMware snapshot tools to create a snapshot of all of the machines in a cluster. However, all machines **must be shut down** before starting the snapshot and must remain shut down until all of the snapshots are complete. This enables any internal cluster communication to be completed before the snapshot is created.

### Info

The services on all members of the cluster are shutdown for the backup and restarted when the backup is complete.

The UI lists the items which will be backed up.

### Destination system time must not be earlier than source when restoring a backup

You must ensure that the current system time on the destination appliance is no earlier than that of the appliance on which the backup was created. If the modification times on the files contained in the backup are later than the system time when they are restored, the backup will hang. To recover at this point you must kill the backup process, correct the time, run `tw_restore --fix-interrupted` then repeat the restore using the `tw_restore` command line utility.

### Backing up and CMDB synchronization

The backup contains the CMDB synchronization configuration. When a host has significantly changed so that its key has also changed, problems can be caused if a backup is restored before the changed host is rediscovered. In this case, on the next CMDB synchronization, duplicate hosts will be created in the CMDB representing the changed host, and the CIs representing the original hosts will never be deleted. To ensure that no duplicate hosts are created, you can delete and then recreate the BMC dataset.

The backup contains the LDAP configuration. If the destination appliance cannot access the LDAP server, you must ensure that a local (non-LDAP) user belonging to the system and public groups is activated and successfully tested on the source appliance before making a backup.

If you choose to exclude the credential vault when backing up an appliance in which CMDB synchronization has been configured, the CMDB Sync page on the restored
appliance displays the "This appliance has not been set up for synchronization with the Atrium CMDB" message. Once the Setup form is complete, filter and blackout window settings are restored.

The appliance backup feature replaces the appliance snapshot that was available in previous releases. On upgraded appliances only, if snapshots are still held in the filesystem, a banner and **Remove Snapshots** button is provided so you can remove the snapshots and release disk space.

**To create a backup of the cluster**

1. From the Appliance section of the **Administration** tab, click **Backup & Restore**. The Appliance Backup page is displayed. This has a panel in which you can configure the backup destination. It also shows details of the cluster size, the size of the backup for this member, and details of the contents of the backup.

   Shows the create backup tab of the appliance backup page for a cluster member.

2. Enter the details for the backup destination. The fields that can be completed are displayed or hidden depending on the backup type selection. Required fields are indicated with a red asterisk.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
   | Backup Type | Select the destination type from the drop down list. This can be one of the following:  
   |             | • On Members — The backup for each machine in the cluster is written to the $TIDEWAY/var/backup directory on that machine. Only one local (On Members) backup can be stored.  
   |             | • SSH — The backup is written to a remote server over ssh.  
   |             | • Windows Share — The backup is written to a Windows share.  
   |             | You cannot create a backup on a Windows share from a FIPS enabled appliance. The mount operation fails and an error message written to syslog. |
   | Notes | A free text area in which you can write notes about the backup. |
   | Directory | The directory into which to write the backup on the remote SSH server. Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup_machine_UUID_inside the specified directory. (SSH only). |
   | Path | The share name and directory name into which to write the backup on a Windows share. Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup_machine_UUID_inside the specified directory. Path syntax is \sharename\directoryname, where sharename is the name of the Windows share, and directoryname is the name of the directory into which to write the backup. (Windows share only). |
### Field Name | Details
--- | ---
Host | The hostname or IP address of the remote server onto which to write the backup (SSH).
Port | The port to which to connect (SSH).
Username | The username to use to connect to the remote server (SSH and Windows share).
Password | The corresponding password (SSH and Windows share).
Verify backup | Select *Verify backup* to verify (md5) that the files in the backup archive are the same as those on the appliance. Shown only after successfully testing the connection for SSH and Windows share backups.
Exclude vault | Select *Exclude vault* to exclude the credential vault from the backup. Appliance UI users are always backed up and restored, regardless of this setting. So, for example, after a restore the password in effect for the system user will be the one from the source appliance. Shown only after successfully testing the connection for SSH and Windows share backups.
Email when complete | Select *Email when complete* and enter an email address if you want an email to be sent automatically when the backup task is completed. Shown only after successfully testing the connection for SSH and Windows share backups.
Test Connection | Click *Test Connection* to ensure that the remote host can be contacted and that the credentials are valid. Shown only for SSH and Windows share backups.

3. Click **Shutdown & Backup** to start the cluster backup operation. You are prompted for confirmation.
4. Click **No** to return to the Appliance Backup page. Click **Yes** to continue and backup the appliance.
   All services are shut down on all members of the cluster for the backup and restarted when the backup is complete. During the backup, a progress window is displayed. A **Cancel** button is also displayed, but only on the UI for the machine from which the backup was started. The **Cancel** button is only enabled at the stages of the backup where it is possible to cancel.

**To restore a backup to the cluster**

1. From the Appliance section of the **Administration** tab, click **Backup & Restore**. The Appliance Backup page is displayed.
2. Click the **Restore Backup** tab.
   The **Restore Backup** tab has a panel in which you can choose the source of the backup, and provides details of the size and contents of the existing local backup.
Shows the restore backup tab of the appliance backup page for a cluster member.

3. Enter the details for the backup source. The fields that can be completed are displayed or hidden depending on the backup type selection. Required fields are indicated with a red asterisk.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup Type</td>
<td>Select the destination type from the drop down list. This can be one of the following:</td>
</tr>
<tr>
<td></td>
<td>• On Members — The backup for each machine is read from that machine's $TIDEWAY/var.backup directory.</td>
</tr>
<tr>
<td></td>
<td>• SSH — The backup is read from a remote server over ssh.</td>
</tr>
<tr>
<td></td>
<td>• Windows Share — The backup is read from a Windows share.</td>
</tr>
<tr>
<td>Directory</td>
<td>The directory from which to read the backup on the remote server (SSH). Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup_machine_UUID_inside the specified directory. Ensure you specify the subdirectory name too. (SSH only).</td>
</tr>
<tr>
<td>Path</td>
<td>The share name and directory name from which to read the backup on the remote server (Windows share). Backups are written into a subdirectory called YYYY-MM-DD_hhmmss_addm_backup_machine_UUID_inside the specified directory. Path syntax is \sharename\directoryname, where sharename is the name of the Windows share, and directoryname is the name of the directory containing the backup. (Windows share only).</td>
</tr>
<tr>
<td>Host</td>
<td>The hostname of the remote server from which to read the backup (SSH).</td>
</tr>
<tr>
<td>Port</td>
<td>The port to which to connect (SSH).</td>
</tr>
<tr>
<td>Username</td>
<td>The username to use to connect to the remote server (SSH and Windows share).</td>
</tr>
<tr>
<td>Password</td>
<td>The corresponding password (SSH and Windows share).</td>
</tr>
<tr>
<td>Preserve Identity</td>
<td>Select the check box to preserve the identity of the current cluster members rather than take on identities from the restored backup. For each member, this consists of:</td>
</tr>
<tr>
<td></td>
<td>• Appliance identity</td>
</tr>
<tr>
<td></td>
<td>• HTTPS configuration</td>
</tr>
<tr>
<td></td>
<td>• Consolidation configuration</td>
</tr>
<tr>
<td></td>
<td>Clear the check box to use the identity from the backup.</td>
</tr>
<tr>
<td>Email when</td>
<td>Select the check box and enter an email address if you want an email to be sent automatically when restoring the backup is complete.</td>
</tr>
<tr>
<td>complete</td>
<td></td>
</tr>
<tr>
<td>Test Connection</td>
<td>When you enter valid connection information, the Test Connection button is enabled. Click this to test the connection to the remote backup server. If the test is successful, and a backup is present, the Remote Backup Details pane displays information on the remote backup.</td>
</tr>
</tbody>
</table>
4. Click **Shutdown & Restore** to start the restore operation.
   You are prompted for confirmation.

5. Click **No** to return to the Appliance Backup page. Click **Yes** to continue and restore the cluster.
   All services are shut down on all members of the cluster for the backup and restarted when the backup is complete.

---

⚠️ **Device package error on restore**

Network device definitions are automatically installed with a TKU. The source and destination appliance must have the same version network device definitions. If you see the following error when restoring, it is important to update the package as described in the error message.

```
An error was reported while installing the device package.
Please correct the error and install the device package manually by running (as root):
   tw_device_import --force -U --oldpackage <rpm file>
```

The device package is located in the `$TIDEWAY/data/installed/tpl` directory.

---

**Shutting down restarting and maintenance mode**

For some maintenance tasks you may need to restart the appliance services, for others you need to reboot or shut down the appliance, and for some you can place the appliance into maintenance mode. For example, you need to:

- Restart services — after making changes to some discovery configuration options
- Shut down — to make hardware changes such as adding disks
- Reboot — after upgrading the BMC Atrium Discovery application software or upgrading the operating system

Maintenance mode is a user mode in which the only users permitted are those who are members of the system group. All users who are not members of this group are logged off the appliance and an explanatory message is displayed.

**To restart services**

You must be logged in as a user who is a member of the system group to shut down the appliance.

1. From the Appliance section of the **Administration** tab, select **Control**.
2. On the Control page, click **Restart Appliance Services**.
3. Confirm the action.
To reboot the appliance
You must be logged in as a user who is a member of the system group to reboot the appliance.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Reboot Appliance.
3. Confirm the action.

To shut down the appliance
You must be logged in as a user who is a member of the system group to shut down the appliance.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Shutdown Appliance.
3. Confirm the action.

To put the appliance into maintenance mode
You must be logged in as a user who is a member of the system group to reboot the appliance.

1. From the Appliance section of the Administration tab, select Control.
2. On the Appliance Control page, click Set Maintenance Mode.
3. Confirm the action.
   All users who are not members of this group are logged off. System group users' screens are refreshed and the Quit Maintenance Mode button is displayed.

⚠️ Maintenance mode is not a single-user mode. If you are performing any tasks that could affect other users (such as appliance backup) you should ensure that you are the only user logged in.
Use the Administration => Security => Active Sessions window to verify this.

⚠️ When non-system users are logged out, the login banner is displayed with an "under maintenance" message. When logging into an appliance that is in maintenance mode, you should ensure that your work does not interfere with that of other logged in users.

In maintenance mode, a flashing banner is displayed at the top of all pages. The flashing banner is a link to the Appliance Control page.

To leave maintenance mode
To leave maintenance mode, click Quit Maintenance Mode.
Shutting down restarting and maintenance in clusters

For some cluster maintenance tasks you may need to restart the services on all machines in the cluster, for others you need to reboot or shut down all machines in the cluster, and for some you can place the cluster into maintenance mode. For example, you need to:

- Restart services — after making changes to some discovery configuration options
- Shut down — to make hardware changes such as adding disks
- Reboot — after upgrading the BMC Atrium Discovery application software or upgrading the operating system

Maintenance mode is a user mode in which the only users permitted are those who are members of the system group. All users who are not members of this group are logged off the cluster and an explanatory message is displayed.

⚠️ Warning

The operations on the Control page in a cluster apply to every machine in the cluster. Individual machines can be controlled from the cluster management page.

To restart cluster services

You must be logged in as a user who is a member of the system group to restart the services on all machines in the cluster.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Restart Cluster Services.
3. Confirm the action.

To reboot the cluster

You must be logged in as a user who is a member of the system group to reboot all of the machines in the cluster.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Reboot Cluster.
3. Confirm the action.

To shut down all machines in the cluster

You must be logged in as a user who is a member of the system group to shut down all machines in the cluster.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Shutdown Cluster.
3. Confirm the action.

To put the cluster into maintenance mode
You must be logged in as a user who is a member of the system group to place the cluster into maintenance mode.

1. From the Appliance section of the Administration tab, select Control.
2. On the Control page, click Set Maintenance Mode.
3. Confirm the action.

All users who are not members of this group are logged off. System group users' screens are refreshed and the Quit Maintenance Mode button is displayed.

- Maintenance mode is not a single-user mode. If you are performing any tasks that could affect other users (such as appliance backup) you should ensure that you are the only user logged in.
  
  Use the Administration => Security => Active Sessions window to verify this.

- When non-system users are logged out, the login banner is displayed with an "under maintenance" message. When logging into an appliance that is in maintenance mode, you should ensure that your work does not interfere with that of other logged in users.

In maintenance mode, a flashing banner is displayed at the top of all pages. The flashing banner is a link to the Control page.

To leave maintenance mode
To leave maintenance mode, click Quit Maintenance Mode.

Performing time synchronization

The Time Synchronization page enables you to synchronize the internal clock of an appliance or virtual machine with a Network Time Protocol (NTP) server on your network or on the Internet or use VMware Tools time synchronization. It configures time synchronization across all the machines in a cluster. Though you can synchronize the time using the command line, see Configuring the NTP client at the command line, the preferred way is to configure through the UI.

An accurate clock is important for the operation of BMC Atrium Discovery. For example, you need to ensure that scheduled discovery runs occur at the correct time, or reconcile discovery access timestamps between multiple machines.

When the NTP daemon starts, it performs an initial synchronization. The initial synchronization can be cancelled. Thereafter, the NTP daemon maintains the correct time with regular checks.
Viewing the Time Synchronization page

To view the Time Synchronization page, from the Appliance section of the Administration tab, click Time Synchronization.

The time synchronization UI.

Fields and links

- **Status** — Displays the status of the time synchronization. It displays No Time Synchronization when both Network Time Protocol and VMware Time Sync are disabled. When you enable VMware Time Sync, the status changes to Using VMware Time Sync. When you enable Network Time protocol, the status changes to Using Network Time Protocol. An additional message Time Synchronization in progress and an option to Cancel Synchronization is displayed when time is first being synchronized using NTP.
- **Network Time Protocol** — Displays whether the NTP time synchronization is currently Enabled or Disabled.
- **NTP Servers** — Displays the list of NTP servers configured for time synchronization.
- **VMware Time Sync** — Displays whether the VMware time synchronization is currently Enabled or Disabled. When Network Time Protocol is enabled, VMware Time Sync gets disabled and this field becomes non-editable. The warning message is displayed as VMware Time Sync will be disabled when NTP is enabled. When Network Time Protocol is disabled, you can optionally re-enable VMware Time Sync.

⚠️ It is recommended that you use NTP instead of VMware Tools time synchronization as NTP provides more precise timekeeping on virtual machines. This is explained in VMware's timekeeping best practices for Linux based Virtual Machines which can be found here under the heading VMware Tools time synchronization configuration.

Actions

- **Restart NTPD** — Select to restart the NTP daemon. The NTP daemon performs the synchronization of the internal time with NTP servers. This option is not displayed when Network Time Protocol is disabled.
To synchronize the time

To synchronize the time of an appliance with the NTP server or across cluster members, perform the following steps on the Time Synchronization page:

1. If the time is not already synchronized, Status displays **No Time Synchronization**. To synchronize using VMware time sync, enable **VMware Time Sync**. To synchronize using NTP, enable **Network Time Protocol**.
2. Enter the details of the NTP server or servers you want to use. You can toggle the server text view between the plain text view and pill view. In the pill view, as you enter text, the UI formats it as a pill (a discrete editable unit) when you enter a space or a comma. According to the text entered, the pill is formatted to represent one of the previous types or presented as invalid. By default, the list is displayed in the pill view.
   - To edit a pill, click the pill body and edit the text.
   - To delete a pill, click the X icon to the right of the pill, or click to edit and delete all of the text.
   - To view the unformatted source text, click the source toggle switch. The source view is useful for copying to a text editor or spreadsheet. Click the source toggle switch again to see the formatted pill view.
   Underneath the entry field is a filter box. Enter text in the filter box to only show matching pills.
3. Select **Apply**. An option to **Cancel Synchronization** is displayed when time is being synchronized using NTP. You can either cancel or continue with the synchronization. When time is synchronized, **Status** is displayed as **Using Network Time Protocol**.
4. If you enable **Network Time Protocol** when **VMware Time Sync** is enabled, the confirmation message is displayed as **Enabling NTP will disable VMware Time Sync on any VMware Virtual Machines in the cluster. Are you sure you want to do this?**. Click **OK** to apply the changes.

Configuring the NTP client at the command line

The NTP client enables the appliance or virtual machine to synchronize its internal clock with an NTP server on your network or on the internet. You can also configure the NTP client through the UI, see **Performing time synchronization**. The preferred way to configure the NTP client is through the UI.

⚠️ **NTP and VMware Tools**
If you are using NTP on a BMC Atrium Discovery Virtual Appliance, you should disable the VMware tools time syncing. This is explained in VMware’s timekeeping best practices for Linux based Virtual Machines which can be found here under the heading VMware Tools time synchronization configuration.

To configure the NTP client you must be logged in to the command line as the root user.

1. **Edit the** `/etc/ntp.conf` **file.**
2. **Search for the lines beginning** `server`.  
   
   ```
   server 0.rhel.pool.ntp.org  
   server 1.rhel.pool.ntp.org  
   server 2.rhel.pool.ntp.org  
   ```

3. **Replace the server entries with the IP address or hostname of the NTP server or servers with which you want to synchronize.** For example:
   
   ```
   server ntp.tideway.com  
   server 172.17.1.24  
   ```

4. **Save the file.**
5. **Configure the NTP client service to start at run level 3 when the appliance boots.** Enter:

   ```
   [root@localhost] # /sbin/chkconfig --levels 3 ntpd on  
   [root@localhost] #  
   ```

6. **Check to ensure that this change has been made correctly.** Enter the following command and ensure that the output is the same as that shown:

   ```
   [root@localhost] # /sbin/chkconfig --list ntpd  
   ntpd    0:off 1:off 2:off 3:on 4:off 5:off 6:off  
   [root@localhost] #  
   ```

7. **Start the service.** Enter:
The NTP client is now running and needs no further attention.

**Rebaseline the appliance after configuring the NTP client**

If you use appliance baseline, you must rebaseline the appliance after making this change. To rebaseline the appliance, as the tideway user, enter the following command:

```
/usr/tideway/bin/tw_tripwire_rebaseline
```

For more information about tripwire and baselining the appliance, see [Baseline configuration](#).

**Message - ntpd is not configured to run at run level 5**

The message `ntpd is not configured to run at run level 5` which is displayed in the appliance baseline window is erroneous. It is displayed when `ntpd` is running.

**Baseline configuration**

Appliance Baseline is a set of conditions that are verified to get a health check of the appliance and decide whether the appliance is healthy, might be tuned for better performance or requires immediate attention. For every problem severity level, depending on the configuration, appliance status changes might launch a notification email, limit network access, or even stop the discovery process. The high level status message is displayed in the Appliance Status box in the dynamic toolbox. Detailed results of the appliance baseline check are available on the Appliance Baseline status page.

- Checks performed
- Viewing the high level appliance status
- Viewing detailed appliance baseline status
- Configuring appliance status options
- Configuring actions on changing appliance status
- Tripwire commissioning and configuration
- Tripwire maintenance
Checks performed

The checks that are performed for each item in the Appliance Baseline Page are described in the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Check Performed</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Configuration</td>
<td>Checks to ensure that the Apache configuration has not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Apache HTTPS</td>
<td>Checks that the HTTPS configuration which allows secure web access (enabled/disabled) on the appliance is the same as that configured in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>AppServer Configuration</td>
<td>Checks to ensure that the application server configuration has not been changed since the last baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>AppServer Start Script</td>
<td>Checks to ensure that the application server start script has not been edited since the last baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>Appliance Configuration Files Tripwire</td>
<td>Checks the tripwire logs to ensure that no appliance configuration files have been added, deleted, or edited since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance Disk Space</td>
<td>Checks the available disk space against the free disk space monitoring settings: the alert and shutdown thresholds.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance Firewall</td>
<td>Checks that the firewall (iptables) configuration matches that recorded in the baseline.</td>
<td>Critical</td>
</tr>
<tr>
<td>Appliance Firewall (IPv6)</td>
<td>Checks that the IPv6 firewall (iptables) configuration matches that recorded in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance HTML Files Tripwire</td>
<td>Checks the tripwire logs to ensure that no HTML files have been added, deleted, or edited since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance Run Level</td>
<td>Checks whether the appliance run level matches that recorded in the baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>Appliance Specification</td>
<td>Checks whether the appliance specification matches that recorded in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance System Files Tripwire</td>
<td>Checks the tripwire logs to ensure that no system files have been added, deleted, or edited since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Appliance eth0</td>
<td>Checks that the eth0 configuration on the appliance is the same as that configured in the baseline. The following items are checked: • Speed • Duplex • Autonegotiation</td>
<td>Minor</td>
</tr>
<tr>
<td>Application Configuration</td>
<td>Checks to ensure that the application configuration has not been changed since the last baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>Application Server</td>
<td>Checks that the UI service is alive.</td>
<td>Critical</td>
</tr>
<tr>
<td>Atrium Credentials</td>
<td>Checks to ensure that the Atrium credentials have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Name</td>
<td>Check Performed</td>
<td>Severity</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Atrium Discovery OS RPM</td>
<td>Checks that the BMC Atrium Discovery Operating System (OS) RPM version number matches that in the baseline.</td>
<td></td>
</tr>
<tr>
<td>Atrium Discovery RPM</td>
<td>Checks that the BMC Atrium Discovery RPM version number matches that in the baseline.</td>
<td>Critical</td>
</tr>
<tr>
<td>Audit Settings</td>
<td>Checks to ensure that the audit settings have not been changed since the last baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>CMDB Sync (Exporter) Service</td>
<td>Checks to ensure that the CMDB Synchronization Export settings have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>CMDB Sync (Transformer) Service</td>
<td>Checks to ensure that the CMDB Synchronization Transformer settings have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>CMDB Sync Blackout Windows</td>
<td>Checks to ensure that the CMDB blackout windows settings have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Cluster Configuration</td>
<td>Checks to ensure that the Cluster Manager configuration matches the baseline configuration.</td>
<td>Major</td>
</tr>
<tr>
<td>Cluster Manager Service</td>
<td>Checks to ensure that the Cluster Manager service settings have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Cluster_manager start script</td>
<td>Checks that the settings in the Cluster_manager start script match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Consolidation</td>
<td>Checks to ensure that the consolidation settings (scanning or consolidation appliance and configured connections including status) have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Crontab</td>
<td>Checks that the cron tab setting on the appliance is the same as that configured in the baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>DDD Removal Blackout Windows</td>
<td>Checks to ensure that the DDD removal blackout windows settings have not been changed since the last baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>DNS Configuration</td>
<td>Checks that the following DNS settings match those in the baseline: • Name servers • Domain</td>
<td>Minor</td>
</tr>
<tr>
<td>DataStore SoftLimit</td>
<td>Checks that the datastore soft limit matches that in the baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>Discovery Configuration</td>
<td>Checks that the Discovery configuration matches that in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Discovery File Content Filters</td>
<td>Checks that the file content filters configured on the appliance match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Discovery Mode</td>
<td>Checks that the Discovery mode (Record/Playback/Normal) matches that in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Discovery Process Filters</td>
<td>Checks that the Discovery Process Filters match these in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Discovery Scripts</td>
<td>Checks that the Discovery commands match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Discovery Service</td>
<td>Checks that the Discovery service is alive.</td>
<td>Critical</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Name</th>
<th>Check Performed</th>
<th>Severity</th>
</tr>
</thead>
</table>
| Discovery Start Script   | Checks that the following settings in the Discovery start script match those in the baseline:  
• Mode - record or playback  
• Log level  
• Pool data expiry time | Minor    |
<p>| Exclusion Ranges         | Checks to ensure that the exclude ranges have not been changed since the last baseline. | Major    |
| Export Adapter           | Checks that the Export Adapter matches those in the baseline.                    | Minor    |
| Export Adapter Configurations | Checks that the Export Adapter configurations matches that in the baseline. | Minor    |
| Export Exporter Configurations | Checks that the Export Exporter configurations matches that in the baseline. | Minor    |
| Export Mapping Sets      | Checks that the Export mapping sets match that in the baseline.                  | Minor    |
| File Export Credentials  | Checks to ensure that the file export credentials have not been changed since the last baseline. | Major    |
| Installed Devices        | Checks whether the installed devices match that recorded in the baseline.         | Minor    |
| Integrations start script| Checks that the settings in the Integration start script match those in the baseline. | Minor    |
| JDBC Credentials         | Checks to ensure that the JDBC credentials have not been changed since the last baseline. | Major    |
| JDBC Drivers             | Checks that the JDBC drivers match those in the baseline.                        | Minor    |
| Login Credentials        | Checks that the Discovery login credentials match those in the baseline.          | Major    |
| Mainframe Credentials    | Checks to ensure that the Mainframe credentials have not been changed since the last baseline. | Major    |
| Miscellaneous Global Settings | Checks to ensure that the miscellaneous global settings have not been changed since the last baseline. | Major    |
| Miscellaneous Local Settings | Checks to ensure that the miscellaneous local settings have not been changed since the last baseline. | Major    |
| Model Service            | Checks that the model service is alive.                                          | Critical  |
| Model Start Script       | Checks to ensure that the model start script has not been edited since the last baseline. | Minor    |
| NTP Configuration        | Checks whether the NTP configuration matches that recorded in the baseline.       | Minor    |
| NTP Running              | Checks whether the NTP status (enabled/disabled) matches that in the baseline.    | Minor    |
|                         | When <code>ntpd</code> is running, the message <code>ntpd is not configured to run at run level 5 is displayed</code> is incorrect and can be ignored. |          |
| Operating System         | Checks whether the OS version matches that in the baseline.                      | Critical  |
|                          | Checks that the pattern configuration matches that in the baseline.              | Major    |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Check Performed</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern Configuration</td>
<td>Checks that pattern definitions match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Modification</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern Definition</td>
<td>Checks that the patterns match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Modifications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern Modification</td>
<td>Checks that the patterns match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Port Scan Settings</td>
<td>Checks that the port scan settings match those in the baseline. The check is</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>performed for each port that is enabled for TCP, UDP, or both.</td>
<td></td>
</tr>
<tr>
<td>Reasoning Service</td>
<td>Checks that the Reasoning service is alive.</td>
<td>Critical</td>
</tr>
<tr>
<td>Reasoning Start Script</td>
<td>Checks that the log level for Reasoning matches that in the baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>Reports Service</td>
<td>Checks to ensure that the Reports service settings have not been changed since</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>the last baseline.</td>
<td></td>
</tr>
<tr>
<td>Reports start script</td>
<td>Checks that the settings in the Reports start script match those in the baseline.</td>
<td>Minor</td>
</tr>
<tr>
<td>SNMP Credentials</td>
<td>Checks that the Discovery SNMP credentials match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>SQL Credentials</td>
<td>Checks to ensure that the SQL credentials have not been changed since the last</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>baseline.</td>
<td></td>
</tr>
<tr>
<td>SSL Appliance Key</td>
<td>Checks that the appliance SSL key file MD5 checksums that match those in the</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>baseline.</td>
<td></td>
</tr>
<tr>
<td>SSL CA Key</td>
<td>Checks that the appliance certificate authority file MD5 checksums that match</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>those in the baseline.</td>
<td></td>
</tr>
<tr>
<td>Security Options</td>
<td>Checks that the security service options match those in the baseline.</td>
<td>Major</td>
</tr>
<tr>
<td>Security Service</td>
<td>Checks that the security service match those in the baseline.</td>
<td>Critical</td>
</tr>
<tr>
<td>Security Start Script</td>
<td>Checks to ensure that the security start script has not been edited since the</td>
<td>Minor</td>
</tr>
<tr>
<td></td>
<td>last baseline.</td>
<td></td>
</tr>
<tr>
<td>Usage Data Collection</td>
<td>Checks that the usage data collection configuration matches that recorded in the</td>
<td>Minor</td>
</tr>
<tr>
<td></td>
<td>baseline.</td>
<td></td>
</tr>
<tr>
<td>WBEM Credentials</td>
<td>Checks to ensure that the WEBEM credentials have not been changed since the last</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>baseline.</td>
<td></td>
</tr>
<tr>
<td>WebLogic Credentials</td>
<td>Checks to ensure that the WebLogic credentials have not been changed since the</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>last baseline.</td>
<td></td>
</tr>
<tr>
<td>Windows Proxy Availability</td>
<td>Checks that all of the Windows proxies respond when pinged.</td>
<td>Info</td>
</tr>
<tr>
<td>Windows Proxy Configuration</td>
<td>Checks that the Windows proxy configuration on the appliance (not the external</td>
<td>Major</td>
</tr>
<tr>
<td></td>
<td>Windows proxies) matches that recorded in the baseline. This includes checking the</td>
<td></td>
</tr>
<tr>
<td></td>
<td>type, version, and position in the Windows proxy order.</td>
<td></td>
</tr>
</tbody>
</table>


### Name | Check Performed | Severity
--- | --- | ---
Windows Proxy Configuration File | Checks to ensure that the `winproxy.conf` file on each connected Windows proxy has not been edited since the last baseline. | 
Windows Proxy Pool Configuration | Checks that the Windows proxy pool configuration on the appliance (not the Windows proxies) matches that recorded in the baseline. | Major
Vault Service | Checks to ensure that the Vault service settings have not been changed since the last baseline. | Major
vCenter Credentials | Checks to ensure that the vCenter credentials have not been changed since the last baseline. | Major
VMware Time Sync | Checks to ensure that the VMware Time Sync settings match those in the baseline. If not running on a VMware platform, the test is skipped. | Major
VMwareTools Running | Checks that VMwareTools is installed and running. If not running on a VMware platform, the test is skipped. If the platform cannot be determined, VMware is be assumed; in this case, if VMwareTools are not required, the test can be disabled. | Major
vSphere Credentials | Checks to ensure that the vSphere credentials have not been changed since the last baseline. | Major

### Viewing the high level appliance status

To view the appliance status, click **Appliance Status** in the dynamic toolbox.

The appliance status list shows the following information:

- **Appliance Name** — the name of the appliance.
- **Appliance Time** — the time read from the appliance’s internal clock.
- **ECA Engines** — the number of ECA engines running. The number of ECA engines affects the maximum number of concurrent discovery requests. For more information, see **Configuring discovery**.
- **Summary link** — a link to the detailed baseline status information. It is labeled with one of the following high level status messages that describe the overall status of the appliance:
  - **No Problems Detected** — The status is green. No problems have been detected.
  - **Status Information Available** — The status is green, but at least one potential problem has been detected which has an information level message.
  - **Minor Problems Detected** — At least one minor problem has been detected with your appliance.
  - **Major Problems Detected** — At least one major problem has been detected with your appliance.
  - **Critical Problems Detected** — At least one critical problem has been detected with your appliance.
Viewing detailed appliance baseline status

To open detailed appliance baseline check results:

- Click Administration in the top navigation bar.
- In the Appliance section, click Baseline status.

The list of baseline checks, their recent results and available actions are displayed.

Configuring appliance status options

You can configure the appliance baseline options such as the recipients of automatic emails, and the messages to be included.

Before you begin

You must have setup email on the appliance before using this feature. See Setting Up Appliance Mail Settings for more information.

To configure appliance status options:

1. From the Appliance section of the Administration tab, select Baseline Status.
   The Appliance Baseline page can also be accessed by clicking Appliance Status in the dynamic toolbox, and then clicking the link in the drop-down list.
2. Click Configure Options.

   The Appliance Baseline Options displays the following options:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email Recipients</td>
<td>The email address or addresses which will be sent an email. This is entered as a single email address or a comma separated list of addresses.</td>
</tr>
</tbody>
</table>
   | Email Subject Template | The template used to create the email subject. By default this is: ADDM Baseline: %(appliance_name)s: %(message)s (%(severity)s) Where:  
   * %(appliance_name) — replaced with the name of the appliance.  
   * %(message) — replaced with the passed message or failed message appropriately.  
   * %(severity) — replaced with the severity of the highest severity check that failed, or OK if the checks all passed. |
If the appliance mail server settings are set to an invalid mail server, configuring baseline to send email introduces a delay of approximately three minutes while the appliance attempts to contact the SMTP server, each time baseline is run. Baseline is run hourly by cron, and can be run manually by a user.

### Configuring actions on changing appliance status

You can configure the actions that will occur when the appliance status changes to critical, major, or minor. Available actions are:

- Send Email
- Restrict Network Access
- Stop Discovery

**To configure actions on changing appliance status:**

1. From the Appliance section of the Administration tab, select **Baseline Status**. The Appliance Baseline page can also be accessed by clicking **Appliance Status** in the dynamic toolbox, and then clicking the link in the drop-down list.
2. Click **Configure Actions**.
3. The is displayed.

   The Appliance Baseline Actions page displays the following options:

### Field Name Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passed Message</td>
<td>The message to include in the email when the test is passed.</td>
</tr>
<tr>
<td>Failed Message</td>
<td>The message to include in the email when the test is failed.</td>
</tr>
</tbody>
</table>

### Services To Allow

Select one or more of the following services to remain open if network access is restricted according to the actions configured. Blocking services such as DHCP or ICMPv6 can prevent the appliance from obtaining an address and it could become unreachable. Also, blocking HTTP, HTTPS, and SSH blocks all remote access to the appliance. In this instance you need to use the system console to fix the problems.

- CLUSTER_MANAGER
- DATASTORE
- DHCP
- DHCPv6
- DNS
- HTTP
- HTTPS
- ICMPv6
- LDAP
- REASONING
- SMTP
- SSH

For example, where a critical problem is detected, you might choose to limit network access to and from the appliance to HTTP or HTTPS only. To do this, select HTTP and HTTPS, ensure the other services are deselected. When a failure occurs that you have configured to restrict network access, the firewall is raised. When the problem is fixed, the firewall is not lowered. To do this you must restart the firewall.
### Field Name Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
</table>
| Actions to take on CRITICAL failure            | Select the actions to take when a CRITICAL failure occurs. The following options are available:  
• Send Email  
• Restrict Network Access  
• Stop Discovery |
| Actions to take on MAJOR failure               | Select the actions to take when a MAJOR failure occurs. The following options are available:  
• Send Email  
• Restrict Network Access  
• Stop Discovery |
| Actions to take on MINOR failure               | Select the actions to take when a MINOR failure occurs. The following options are available:  
• Send Email  
• Restrict Network Access  
• Stop Discovery |
| Actions to take on INFO only                   | Select the actions to take when an INFO failure occurs. The following options are available:  
• Send Email  
• Restrict Network Access  
• Stop Discovery |
| Actions to take on SUCCESS                     | Select the action to take when there are no failures. The following option is available:  
• Send Email |

### Tripwire commissioning and configuration

Tripwire is a third-party software tool that monitors a given set of configuration, system, and source files on an appliance. For further information about Tripwire, see: [http://sourceforge.net/projects/tripwire/](http://sourceforge.net/projects/tripwire/). Tripwire is installed by the kickstart process but is not commissioned. When Tripwire has been commissioned, it is run hourly. You can also run it manually, see Running Tripwire checks manually for more information.

The Tripwire reports are stored in the following directory: `/usr/tideway/var/tripwire/report`  
You must create this directory if it does not exist. As the tideway user, enter the following command:

```
mkdir -p /usr/tideway/var/tripwire/report
```

### Adding tripwire configuration to appliance backup

The tripwire configuration is not included in an appliance backup by default. If you want to include it, add the following to the `$TIDEWAY/etc/backup_config.xml` file.
The tripwire directory is archived into the backup directory in a file called `addm_tripwire_etc.tgz`. The archive is not restored when the backup is restored but can be copied manually onto the restored appliance and recommissioned using the Commissioning Tripwire passkeys procedure.

**Commissioning Tripwire passkeys**

Commissioning Tripwire passkeys is a one-off procedure. You must be able to log in as the root user to complete Tripwire passkeys commissioning.

1. **Log in as the root user.**
   - The default Tripwire policy file is `/usr/tideway/etc/twpol.txt`.
2. **Edit the file and enter the hostname of the appliance (as returned by the `hostname` command), replacing `localhost`.**
   - An excerpt of the file is shown below:

   ```
   @@section GLOBAL
   TWROOT="/usr/tideway/tripwire/sbin";
   TWBIN="/usr/tideway/tripwire/sbin";
   TWPOL="/usr/tideway/tripwire/etc";
   TWDDB="/usr/tideway/tripwire/var/lib";
   TWSKEY="/usr/tideway/tripwire/etc";
   TWLKEY="/usr/tideway/tripwire/etc";
   TWREPORT="/usr/tideway/var/tripwire/report";
   ARCH="x86_64";
   HOSTNAME="localhost";
   ```

3. **If you want to monitor any additional files, add the full path to that file to the policy file.**
4. **If you want to monitor any additional directories, add the full path to that directory to the policy file.**
5. **Copy the `/usr/tideway/etc/twpol.txt` file to `/usr/tideway/tripwire/etc/twpol.txt`, overwriting the existing file.**
6. **Run the following command which sets up the initial database and passwords allowing changes to the Tripwire configuration**
   
   ```
   /usr/tideway/tripwire/sbin/tripwire-setup-keyfiles
   ```
7. You are prompted to create a site and a local password. Record these passwords or you will need to reinstall the Tripwire database.
   The local password is required to remove Tripwire violations.
   The site password is required to update the Tripwire policy file.
8. You are prompted to sign the configuration file `twcfg.txt` and the policy file `twpol.txt`.
9. Change the ownership and permissions of the `twcfg.txt` and the `twpol.txt` files to the `tideway` user. Enter the following commands:

   ```
   cd /usr/tideway/tripwire/
   chown tideway:tideway etc
   chmod 750 etc
   cd etc
   chown tideway:tideway twcfg.txt twpol.txt
   chmod 640 twcfg.txt twpol.txt
   ```

Initializing the Tripwire database

Initializing the Tripwire database is a one-off procedure. This procedure should be performed as the `tideway` user.

1. The Tripwire database must be initialised with the contents of the Tripwire policy file.
2. Run the following command to initialize the Tripwire database:

   ```
   sudo /usr/tideway/tripwire/sbin/tripwire --init
   ```

3. Run the following command to rebaseline the Tripwire database:

   ```
   /usr/tideway/bin/tw_tripwire_rebaseline
   ```

   An error is reported as a database backup file is created.
4. Run the following command again to rebaseline the Tripwire database:

   ```
   /usr/tideway/bin/tw_tripwire_rebaseline
   ```

   This time, no errors are reported as no files have been added. The tripwire database is now initialised and baselined.
Initial appliance baseline configuration

When you have freshly configured the tripwire database, the appliance baseline must be updated to ensure that the correct status is shown in the user interface.

**Warning**

This causes all of the appliance baseline checks to be reset. Make sure that all existing baseline failures are addressed.

1. Run `/usr/tideway/bin/tw_baseline` or click Check Baseline Now in the user interface to execute all the baseline tests.
2. Verify that only tripwire related tests are failing. Tripwire test names end with "tripwire".
3. Update the tripwire report and then update the appliance baseline as follows:

```
sudo /usr/tideway/tripwire/sbin/tripwire --check > /usr/tideway/var/tw_tripwire.txt
/usr/tideway/bin/tw_baseline --rebaseline
```

The appliance status is updated, and tripwire commissioning is now complete.

Tripwire maintenance

Updating after a violation

When you use the `tw_tripwire_rebaseline` utility to rebaseline the Tripwire database, you accept that all files that are being monitored are correct. This procedure should be performed as the tideway user. To update the Tripwire database after an error:

1. Check the items that are reported in the violation report and ensure that the reported changes are what you expected.
2. Run the following command:

```
/usr/tideway/bin/tw_tripwire_rebaseline
```

Updating the Tripwire policy file

Sometimes you must update the Tripwire policy file. For example:

- An EFix being applied
- A full system upgrade
- Appliance relocation or change of IP Address
- Files changing too frequently and creating false positive alerts
Edit `/usr/tideway/tripwire/etc/twpol.txt` and make the necessary changes. Save the file using the same name.
Clear all violations before updating the Tripwire policy file by rebaselining the Tripwire database. The system must be in a known good state to update the policy database. This procedure should be performed as the tideway user.

1. Run the following command to rebaseline the Tripwire database:

   `/usr/tideway/bin/tw_tripwire_rebaseline`

2. Run the following command (on one line) to update the Tripwire policy file:

   ```bash
cd /usr/tideway/tripwire/etc/
   sudo /usr/tideway/tripwire/sbin/tripwire --update-policy twpol.txt
   ```

   You need both the local and site password for this operation.

3. Check that the update has been performed correctly. Enter:

   ```bash
   sudo /usr/tideway/tripwire/sbin/tripwire --check
   ```

4. Run the following command to rebaseline the Tripwire database:

   `/usr/tideway/bin/tw_tripwire_rebaseline`

For more information about the `tw_tripwire_rebaseline` utility, see `tw_tripwire_rebaseline`.

**Running Tripwire checks manually**

By default, Tripwire is run hourly and the output is written to the `tw_tripwire.txt` file. If a deviation from the baseline has been detected, the `tw_tripwire.txt` file is updated with the details. The monitor which sets the appliance status in the user interface checks the `tw_tripwire.txt` file hourly and sets certain restrictions if configured.

If you have rebaselined the Tripwire database, you should run the following commands to ensure that the correct status is shown in the user interface.

```bash
sudo /usr/tideway/tripwire/sbin/tripwire --check > /usr/tideway/var/tw_tripwire.txt
/usr/tideway/bin/tw_baseline --rebaseline
```
Managing standard data

This section explains how to set up and view standard data on the BMC Atrium Discovery system. You must set up details of the specific data required by your IT organization, such as the locations, product groupings, and status values that you use. Known as standard data categories, they are typically set up at initial deployment, but you can change them later if necessary. Additional standard data that you can view from the Setup menu includes the datastore partitions (discrete areas of data in the datastore) and the system taxonomy, which includes details of all the kinds of data object in the BMC Atrium Discovery system and the relationships between them.

- Setting up standard data categories
- Purging history and destroyed nodes
- Viewing the system taxonomy

Setting up standard data categories

The standard data categories define areas of basic data, specific to your organization, that cannot be obtained from data discovery. This includes the locations, product groupings, status values and so on that are relevant to your organization. You would normally set up all this data when the system is first configured for your organization, but you can update it at any time if required.

Viewing the standard categories

To view the standard data categories, click the Custom Categories icon in the Model section of the Administration tab. The Custom Categories page is displayed.

Accessing data in the standard categories

1. The Custom Categories page shows tabs representing the categories of data that you can manage.
2. Click the category tab for the data that you want to view, create or edit. The categories are:

<table>
<thead>
<tr>
<th>Category</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttachmentCategory</td>
<td>Groupings of files attached to objects. See #Managing Attachment Categories.</td>
</tr>
<tr>
<td>Family</td>
<td>Groupings of Business Application Instances that are specific to your organization. See #Managing Product Families.</td>
</tr>
<tr>
<td>Lifecycle Status</td>
<td>Status values that can be associated with the IT components managed in your organization. See #Managing Lifecycle Status Values.</td>
</tr>
<tr>
<td>Location</td>
<td>Physical locations in your organization which can be associated with a managed element or a person. See #Managing Locations.</td>
</tr>
<tr>
<td>Organisational Unit</td>
<td></td>
</tr>
</tbody>
</table>
2.

1. Logical divisions of your organization, such as a department or a business unit in your organization. See #Managing Organisational Units.

Recovery Time

Valid recovery times that can be applied to the IT components that can be managed in your organization. See #Managing Recovery Time Values.

Each tab for the standard data shows the data types that are configured on the appliance, along with abbreviations, descriptions, whether that type is active or not, arrow keys to reorder the types, whether the type is standard in BMC Atrium Discovery or user defined, and a Delete button to delete that data type. The page also has a New button which enables you to create a new type, depending on the page you are viewing.

Managing attachment categories

BMC Atrium Discovery users can associate any file - such as a document, a spreadsheet, or a diagram - with an object in the BMC Atrium Discovery datastore in order to provide additional information. You can associate an attachment with a Software Instance, Business Application Instance, or Host. Files can be of any type and are not validated in any way by the BMC Atrium Discovery system. Attached files are uploaded to the BMC Atrium Discovery datastore so that they can be accessed easily for disaster recovery purposes.

When you select the Attachment Category page all existing categories are listed. From this page you can create a new category or display an existing one in detail.

Creating a new attachment category

1. From the Custom Categories page, select the Attachment Category tab. The existing attachment categories are displayed.
2. Click the New Attachment Category button. The Create Attachment Category page is displayed.
3. Complete the details of the attachment category.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of attachment category.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Abbreviation of category name.</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of attachment category.</td>
</tr>
<tr>
<td>Active</td>
<td>Yes if this category is currently active, No if it is not active.</td>
</tr>
<tr>
<td>Attachments</td>
<td>Relationship that defines the actual attachment objects in this category. This information is normally set up when the documents are attached to the objects, but you can search for and select one or more Attachment objects.</td>
</tr>
</tbody>
</table>

4. Click Apply to save the attachment category details.
Viewing details of an attachment category

1. From the Custom Categories page, select the Attachment Category tab. The existing attachment categories are displayed.
2. Click an entry to display the View Object page showing full details of the selected category.
   - To delete the selected category, click Delete.
   - To view audit details of the category, click History.

From the Attachment Category you can also view relevant attachments.

Managing product families

Product Families define the groupings of Business Application Instances that are specific to your organization.
When you select the Product Family page all existing families are listed. From this page you can create a new family or display an existing one in detail.

Creating a new product family

1. From the Custom Categories page, select the Families tab. The existing families are displayed.
2. Click New Family button. The Create Family page is displayed.
3. Complete the Attributes and Relationships of the product family.
   - **Field Name**
     - **Details**
       - **Name**: Name of product family.
       - **Abbreviation**: Abbreviation of product family name.
       - **Description**: Free-text description of product family.
       - **Active**: Yes if this product family is currently active, No if it is not active. Defaults to Yes.
       - **Standard**: Yes if this product family is standard in your organization; No if it is not active. Defaults to Yes.
       - **Applications in Family**: Relationship that defines the Business Applications that are included in this product family. This information is normally set up when you set up the Business Applications, or can be set by a pattern, or you can search for and select one or more applications from here.
       - **Family Owner**: Relationship that defines the ownership of this product family. Search for and select the appropriate Person.
4. Click Apply to save the product family details.

Viewing details of a product family

1. From the Custom Categories page, select the Families tab. The existing families are displayed.
2. Click an entry to display the View Object page showing full details of the selected family.
   - To delete the selected family, click Delete.
   - To view audit details of the family, click History.
Managing lifecycle status values

A lifecycle status object defines a specific status value that can be applied to the IT components managed in your organization. You can assign a status value to each managed element (for instance, each Software Product, Business Application Instance and Host).

You would normally set up all the lifecycle status values appropriate to your organization when you first set up the system, but you can update them at any time.

From the Lifecycle Status page all lifecycle status values are listed. From this page you can create a new lifecycle status or display an existing one in detail.

Creating a new lifecycle status value

1. From the Custom Categories page, select the Lifecycle Status tab. Existing lifecycle status values are displayed.
2. Click New Lifecycle Status button. The Create Lifecycle Status page is displayed.
3. Complete the Attributes and Relationships of the host type.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this lifecycle status value.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Abbreviated life-cycle status name.</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of life-cycle status.</td>
</tr>
<tr>
<td>Active</td>
<td>Yes if this lifecycle status is currently active, No if it is not active.Defaults to Yes.</td>
</tr>
<tr>
<td>Withdrawn State?</td>
<td>Yes if this lifecycle status value is Withdrawn, No if it is not. Defaults to No.</td>
</tr>
</tbody>
</table>

4. Complete the Relationships of the lifecycle status.
5. Click Apply to save the lifecycle status details.

Viewing details of a lifecycle status

1. From the Custom Categories page, select the Lifecycle Status tab. Existing lifecycle status values are displayed.
2. Click an entry to display the View Object page showing full details of the selected lifecycle status.
   - To delete the selected lifecycle status, click Delete.
   - To view audit details of the lifecycle status, click History.

Managing locations

A Location object defines a physical location in your organization which can be associated with a managed element or a person. You can create various levels of locations - for example, you can model buildings in sites, or rooms on floors.
You would normally set up all the location values appropriate to your organization when you first set up the system, however, you can update them at any time.

If you need to link locations to other nodes, you can do this by means of patterns. A template pattern is available to do this. For information on template patterns, see Pattern templates. For information on modeling your business applications using patterns, see the Tideway web site and click the Community tab and follow the Documentation Resources link.

From the Location page all locations are listed. From this page you can create a new location or display an existing one in detail.

Creating a new location

1. From the Custom Categories page, select the Locations tab. Existing location values are displayed.
2. Click New Location. The Create Location page is displayed.
3. Complete the Attributes and Relationships of the location.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of location.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Abbreviated name of location.</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of location.</td>
</tr>
<tr>
<td>Active</td>
<td>Yes if this location is currently active, No if it is not active. Defaults to Yes.</td>
</tr>
<tr>
<td>Type</td>
<td>Select type of location (Major or Minor) from drop-down list.</td>
</tr>
<tr>
<td>Address</td>
<td>Address of location.</td>
</tr>
<tr>
<td>Phone</td>
<td>Telephone number of this location.</td>
</tr>
<tr>
<td>Subsidiary Locations</td>
<td>Relationship that defines other locations that are subsidiary to this one. Search for and select one or more locations.</td>
</tr>
<tr>
<td>Parent Location</td>
<td>Relationship that defines the parent location of this one. Search for and select the appropriate location.</td>
</tr>
<tr>
<td>Hosts at this Location</td>
<td>Relationship that defines the host systems that are in this location. Can be set by a pattern, or you can search for and select one or more hosts.</td>
</tr>
<tr>
<td>Subnets at this Location</td>
<td>Relationship that defines the subnets that are in this location. Can be set by a pattern, or you can search for and select one or more subnets.</td>
</tr>
<tr>
<td>Applications at this Location</td>
<td>Defines instances of Business Applications in this physical location. Search for and select one or more applications.</td>
</tr>
</tbody>
</table>

4. Click Apply to save the location details.

Viewing details of a location

1. From the Custom Categories page, select the Locations tab. Existing location values are displayed.
2. Click an entry to display the View Object page showing full details of the selected locations.
   - To delete the selected location, click Delete.
To view audit details of the location, click History.

Managing organizational units

An Organizational Unit object defines a logical division of your organization, such as a department or a business unit in your organization. You can create various levels of organizational unit - for example, you can model teams in functional areas. Managed elements in your organization (Software Product, Business Application and Host) can be associated with particular organizational units.

From the Organisational Unit page all organizational units are listed. From this page you can create a new organizational unit or display an existing one in detail.

Creating a new organizational unit

1. From the Custom Categories page, select the Organisational Units tab. Existing organizational units are displayed.
2. Click New OrganisationalUnit. The Create Organisational Unit page is displayed.
3. Complete the Attributes and Relationships of the organizational unit.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of organizational unit.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Abbreviated form of organizational unit name.</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of organizational unit.</td>
</tr>
<tr>
<td>Active</td>
<td>Yes if this organizational unit is currently active, No if it is not active. Defaults to Yes.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of organizational unit.</td>
</tr>
<tr>
<td>Subsidiary Units</td>
<td>Relationship that defines other organizational units that are subsidiary to this one. Search for and select one or more organizational units.</td>
</tr>
<tr>
<td>Parent Unit</td>
<td>Relationship that defines the parent organizational units of this one. Search for and select the appropriate organizational unit.</td>
</tr>
</tbody>
</table>

4. Click Apply to save the organizational unit details.

Viewing details of an organizational unit

1. From the Custom Categories page, select the Organisational Units tab. Existing organizational units are displayed.
2. Click an entry to display the View Object page showing full details of the selected organizational unit.
   - To delete the selected organizational unit, click Delete.
   - To view audit details of the organizational unit, click History.

⚠️ Searching for Organizational Units
To search for Organizational Units with a Generic Search Query, you must use the following:

```
SEARCH OrganisationalUnit
```

Note the 's' instead of 'z' in OrganisationalUnit.

Managing Recovery Time Values

A Recovery Time object defines a valid recovery time that can be applied to the IT components that can be managed in your organization. You can assign a recovery time to each managed element in your organization (Software Product, Business Application and Host).

You would normally set up all the recovery time values appropriate to your organization when you first set up the system, but you can update them at any time. From the Recovery Time page all recovery time values are listed. From this page you can create a new recovery time or display an existing one in detail.

Creating a new recovery time

1. From the Custom Categories page, select the Recovery Times tab. Existing recovery times are displayed.
2. Click New Recovery Time. The Create Recovery Time page is displayed.
3. Complete the Attributes and Relationships of the recovery time value.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of this recovery time value.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Abbreviation of recovery time value.</td>
</tr>
<tr>
<td>Description</td>
<td>Free-text description of recovery time value.</td>
</tr>
<tr>
<td>Active</td>
<td>Yes if this recovery time value is currently active. No if it is not active. Defaults to Yes.</td>
</tr>
</tbody>
</table>

4. Click Apply to save the recovery time details.

Viewing details of a recovery time

1. From the Custom Categories page, select the Recovery Times tab. Existing recovery times are displayed.
2. Click an entry to display the View Object page showing full details of the selected recovery time.
   - To delete the selected recovery time, click Delete.
   - To view audit details of the recovery time, click History.
Purging history and destroyed nodes

History
History information about each node and relationship is recorded in the datastore. This audit trail includes details of the changes made, the date and time of the changes and the user who made the changes. You can view the history of each object in the datastore and you can search for and display details of destroyed objects.

In previous releases a purge history script was provided enabling you to purge the history manually. From BMC Atrium Discovery 9.0 this can be done automatically. By default, history is not purged. For information on how to enable this and specify an age above which history data is purged, see configuring model maintenance settings.

To ensure that the historical information is self-consistent, the creation time record for each node is maintained, and all state changes that occurred between the creation time and the purge time are rolled up into a single state change entry.

Destroyed nodes
When a node is destroyed, it is marked as destroyed but remains in the datastore for a fixed period of time, after which it is automatically purged. For information on how to specify a time after which destroyed nodes are purged, see configuring model maintenance settings.

Viewing the system taxonomy
The system taxonomy defines all kinds of data objects in the system and the relationships between them. The BMC Atrium Discovery Administrator can examine all of these objects, including their attributes, roles and relationships.

None of this information can be changed from the BMC Atrium Discovery user interface.

Viewing the system taxonomy
1. Click Administration, and then click View Taxonomy.
   All base taxonomy objects are listed, along with the following information:
   - **Object Name** — the name of the object kind. The superclass of this object is displayed in brackets as a hyperlink.
   - **Description** — a description of this object type.
   - **Subclasses** — subclasses of this object type.
   - **Attributes** — attributes of this object type.
   - **Known relationships** — known relationships for this object. The object to which this relationship refers is displayed as a hyperlink at the end of the relationship name.
You can download the base taxonomy in XML format. To do this, right-click the appropriate link and select **Save Target As**....

The Files tab shows the taxonomy files that are installed on the appliance. Further details of the data model can be found in the BMC Atrium Discovery taxonomy. To view the taxonomy on the BMC Atrium Discovery UI, from the **Administration** tab, click **View Taxonomy**. A link to the current, full-size version of the diagrams below is also available in the online documentation.

⚠️ **Taxonomy limitation**

The relationships displayed in the taxonomy are not exhaustive. That is, all possible relationships are not displayed in the taxonomy.

For example, the following containment relationships are valid:

- Group:Container:Containment:ContainedItem:SoftwareInstance
- Group:Container:Containment:ContainedItem:Host

The following diagrams show the BMC Atrium Discovery Default Data Model with its different nodes, attributes and relationships between different parts of BMC Atrium Discovery. It is split across the following diagrams:

- **Main Diagram**—contains all the core entities used in the modeling and discovery of hosts.
- **Network and Printer Diagram**—contains entities used in the modeling and discovery of network devices and printers.
- **Mainframe Diagram**—contains entities used in the modeling and discovery of mainframe computers.
- **Storage and Load Balancers**—contains entities used in the modeling and discovery of storage entities and load balancers.

It is important to note that the model itself is not in separate sections, the separate diagrams exist in order to more clearly convey information.
Color-Code Key

- Inferred nodes and connection links— **Blue**.
- DDD nodes and connection links— **Green**.
- Knowledge (Pattern Management) nodes and connection links— **Pink**.
- Provenance Relationship connection links— **Red**.
- Auxiliary nodes and connection links—Brown (used internally by BMC Atrium Discovery).

You can also download a pdf file [here](#) which includes all of the diagrams above.

⚠️ The PNG images are static and represent the latest taxonomy. They do not reflect any on-site changes that might have been made to the taxonomy on your appliance.
Storage and modification of the taxonomy

Taxonomy definitions are configured using xml files which are held in the following directories:

1. /usr/tideway/data/installed/taxonomy/
2. /usr/tideway/data/custom/taxonomy/

The directories are parsed in the order given (installed before custom), and the files contained in these directories are parsed in alphabetical order, with numbers before letters. This order is important as later definitions for the taxonomy override those loaded earlier.

The standard base taxonomy file is contained in /usr/tideway/data/installed/taxonomy/00taxonomy.xml

To add additional details to a node, we recommend using a detail node rather than extending the taxonomy. For more information, see Detail node.

Managing clusters

This section describes clusters and how to set up and use a clustered BMC Atrium Discovery system. A cluster consists of two or more coordinated BMC Atrium Discovery machines, one of which is in control of the group and is referred to as the coordinator. You can add new machines to a cluster or remove machines from a cluster, without interrupting normal operations. Clusters can be configured with fault tolerance meaning that if a machine fails, it can be removed and replaced, without any consequent loss of data, and without interrupting normal operation.

Do not change the CMDB synchronization configuration at the same time as changing the cluster configuration.

- Changing the CMDB sync configuration means, adding connections, removing connections, and starting or stopping a resync.
- Changing the cluster configuration means adding members, removing members, moving the coordinator, or changing fault tolerance.

Any cluster management and configuration is undertaken from the Cluster Management page of any of the members of a cluster. To access the Cluster Management page:

On the Administration tab, click Cluster Management in the Appliance section.

This example page Cluster Management page is from the coordinator's UI and shows a cluster of three machines where all machines are operating normally.
The cluster management tasks that you can undertake from this page are described in the following topics:

- Creating a cluster
- Adding a machine to an existing cluster
- Changing the machine that is the coordinator
- Changing the address of a machine
- Removing a machine from a cluster
- Reverting a cluster member into a standalone machine

A cluster is not removed, it stops being a cluster when the last member is removed from the coordinator.

Information displayed on the Cluster Management page

This example Cluster Management page shows multiple pending changes. The machine that is leaving the cluster is displayed in **Pending Changes** as it is waiting to leave, and it is also displayed in **Current Members** as it has not yet left the cluster. It will only do so when the changes are committed. When the machine is removed, there will be too few members in the cluster to maintain fault tolerance. Consequently, one of the pending changes is disabling fault tolerance.

You can add multiple changes to the **Pending Changes** section and commit them all at once. Many cluster changes require a rebalance when committed individually, so committing multiple changes at once avoids the need for multiple rebalances.
The Cluster Management page consists of the following basic sections:

- A cluster control section
  - **Shutdown Cluster** button — shuts down all machines in the cluster.
  - **Reboot Cluster** button — reboots all of the machines in the cluster.
  - **Restart Cluster Services** button — restart the services on all machines in the cluster.
  - **Enable Maintenance Mode** button — places the cluster into maintenance mode.

  These buttons duplicate the cluster control functionality available on the Control page.

- An overview information section
  - **Name** — the name of the cluster you are viewing.
  - **Alias** — an alias for the cluster, primarily intended for use with load balancers. For example, a cluster has members called member1, member2 and member3. DNS is configured to resolve the name cluster100 to a load balancer. The load balancer is configured to share requests to cluster100 with the following hosts: member1, member2 and member3. In this case, the cluster alias is cluster100.
  - **Summary** — the status of the cluster, whether the cluster is operating normally, or whether any tasks such as rebalancing are in progress, and a measure of the task's progress.
  - **Coordinator** — whether the current machine is the coordinator or not; if not, a link is provided to the Cluster Management page on the coordinator UI.
- **Fault Tolerance** — whether fault tolerance is enabled or not and a button to enable or disable it depending on the current setting.

- The **Current Members** panel which contains rows with detailed information on the machine or machines currently in the cluster and a mass actions drop down which enables you to remove any selected non-coordinator machine. The mass actions drop down is disabled when the cluster is **rebalancing**. The information on each cluster member includes:
  - **Type** — Whether the machine is a coordinator or a member.
  - **Volume** — The amount of free disk space available in the /usr partition.
  - **Health** — Whether or not there are any issues with the machine.
  - **Activity** — Whether or not the machine is operating normally.
  - **Last Contact** — When the machine last responded to communication from the coordinator.

Each row contains an Actions drop down list enabling you to:

- **Change Address** — change the IP address or hostname used to communicate the machine. You can use this if a machine is assigned a new IP address, or you need to communicate using its hostname.
- **Ping** — Pings the machine to ensure that it can be contacted. On a successful ping, the information is refreshed.
- **Remove** — Remove the non-coordinator machine from the cluster. On selecting this, an entry row is placed in the **Pending Changes** panel. Removing a healthy machine from the cluster will reset that machine to the default configuration.
- **Make coordinator** — Makes this machine the coordinator. See Changing the machine that is the coordinator for more information.
- **Restart Services** — Restarts the services on this machine. This option is only available when fault tolerance is enabled.
- **Reboot** — Reboots this machine. This option is only available when fault tolerance is enabled.
- **Shutdown** — Shuts down this machine. This option is only available when fault tolerance is enabled.

Additional sections are displayed as appropriate. For example, the **Previous Members** panel is only displayed when a machine has been removed from the cluster:

- A results banner with the result of the most recent operation.
- A **There are pending cluster changes** list which includes buttons to Commit or Discard changes.
- A **Pending Changes** panel which contains rows with detailed information on the machine or machines in the pending changes list. Each row contains an Actions drop down list enabling you to remove the row from the panel.
- A **Previous Members** panel which contains rows with detailed information on the machine or machines that have been members of the list. Each row contains an Actions drop down list enabling you to remove the row from the panel.
Cluster overview

From this release, all instances of BMC Atrium Discovery are referred to as machines. In previous releases the term appliance was used to refer to a BMC Atrium Discovery instance.

Clusters

A group of coordinated BMC Atrium Discovery machines is called a cluster. A cluster consists of two or more BMC Atrium Discovery machines, one of which is in control of the group and is referred to as the coordinator.

All machines are capable of acting as:

- a standalone BMC Atrium Discovery machine
- a member of a cluster
- a cluster coordinator

A cluster is not removed, it stops being a cluster when the last member is removed from the coordinator.

What is a cluster for?

Clusters are designed for organizations with very large IT infrastructures where the performance required is greater than that of a single BMC Atrium Discovery machine. The problem and the solution are described in Big Discovery.

Cluster recommendations

The individual machines that make up a cluster are intended to be in the same location. The data in the datastore is distributed across all machines so there is significant traffic between cluster members. Putting the cluster machines into different locations increases the communication overhead and slows cluster operations. Clustering is designed with the intention that machines are located on the same fast LAN.

Clusters should be built from machines of similar specifications and performance. The datastore is distributed over multiple machines, with each machine holding a roughly equal share of the stored data. The total capacity of the distributed datastore is therefore limited by the machine with the lowest disk capacity allocated to the datastore. The performance of the cluster is affected by the lowest performing machine, but as the discovery workload is shared, the impact is not a limit, but a reduction in overall performance.

Naming of machines in a cluster

The individual machines in a cluster are named according the the following scheme:

- Coordinator – clusternameme-01
- First member added – clusternameme-02
- Next member added – clusternameme-03
- And so on
Cluster communications

Clustered machines communicate on the following TCP ports:

- 25030 — cluster management
- 25031 — datastore communication
- 25032 — reasoning communication

See Firewall Port Summary for more information.

Clusters and JDBC drivers

In a cluster, JDBC drivers for querying databases using integration points that are uploaded through the Administration > JDBC Drivers page are synchronized across the cluster automatically. Where you create a new driver the driver JAR and properties files must be copied manually. You must restart the tideway services on all machines in the cluster before using the drivers.

When replacing a JDBC driver with a newer version through the Administration > JDBC Drivers page on the coordinator UI, the changes are synchronized across the cluster, but you must also restart the tideway services on all machines in the cluster.

Fault tolerance

Clusters have been introduced to BMC Atrium Discovery version 10 to improve discovery, and searching and reporting performance. Clusters work by sharing data and the work in processing that data across each machine in the cluster. As cluster size increases, so does the likelihood of one or more of the machines in the cluster experiencing some sort of hardware failure. To prevent data loss in the event of hardware failure, fault tolerance has been built into BMC Atrium Discovery.

To prevent data loss when a machine fails, you can store copies of your data on more than one machine. If every bit of data is stored on two different machines, then any one machine can fail with no resultant loss of data. If every bit of data is stored on three machines, then two machines can fail without data loss. The number of copies of the data governs the number of machines that can fail in the cluster before failures cause data loss. The number of copies, or the replication factor, is configured automatically, as follows:

<table>
<thead>
<tr>
<th>Cluster size</th>
<th>Replication factor</th>
<th>Number of failures tolerated</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 (not fault tolerant)</td>
<td>0 (not fault tolerant)</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0 (not fault tolerant)</td>
</tr>
<tr>
<td>3 to 8</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>9 to 15</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>16 and over</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>
In the case of the fault tolerant cluster of two, the cluster can experience the failure of a machine without losing data. However, the cluster cannot continue to operate as the coordinator can no longer duplicate data the required number of times (the replication factor), in this case two, as there is only one working machine. To make the cluster operational again, you must add another member, either fix the failed machine, or force remove it from the cluster.

If you enable fault tolerance, the cluster will survive the loss of a machine without losing data, or interruption to discovery, and the UI enables you to control the cluster just as before the failure. If the failure was transient, or the machine was repaired, when it returns it starts to process data again, perform discovery, and catch up with the rest of the cluster. If the failure of the machine is permanent, you can use the cluster manager to remove the machine from the cluster and the cluster will redistribute (rebalance) the data over the remaining members. If you replace the machine then a new copy of the data held on the failed machine is copied to the new one.

Fault tolerance works by storing data on multiple machines. As a consequence of having multiple copies, the total storage capacity of the cluster is reduced. Additionally, the overhead of writing, tracking, and searching through multiple copies reduces the overall performance of the cluster, relative to the same cluster with fault tolerance disabled.

**Enabling and disabling fault tolerance**

Fault tolerance can be enabled when creating a cluster by selecting the fault tolerance checkbox in the create cluster dialog.

**To enable fault tolerance in an existing cluster**

From the Fault Tolerance section of the cluster manager, click the **Enable Fault Tolerance** button.

The summary shows the progress of the rebalancing operation. For a large system containing a large amount of data, this may take some time. However, the system remains fully usable throughout.

**To disable fault tolerance in an existing cluster**

From the Fault Tolerance section of the cluster manager, click the **Disable Fault Tolerance** button.

The summary shows the progress of the rebalancing operation. For a large system containing a large amount of data, this may take some time. However, the system remains fully usable throughout.

**Creating a cluster**

You can create a cluster from any BMC Atrium Discovery machine. The machine from which you create the cluster becomes the coordinator for that cluster. If an appliance has any changes from the default configuration, you can create a cluster with it as the coordinator, but you cannot add it to an existing cluster.
To create a cluster:

1. On the Administration tab, click Cluster Management in the Appliance section.

![Cluster Management page](image1)

Shows the Cluster Management page for a machine in a default configuration.

2. Click Create Cluster.

The Create Cluster dialog displays.

![Create Cluster dialog](image2)

Shows the dialog for creating a cluster. The dialog uses pills.

3. Enter the following information:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster name</td>
<td>The name of the cluster.</td>
</tr>
<tr>
<td>Cluster alias</td>
<td>The Cluster Alias is an optional name for referring to all the machines in the cluster, such as when the machines are accessed through a load balancer. For example, a cluster has members called member1, member2 and member3. DNS is configured to resolve the name cluster100 to a load balancer. The load balancer is configured to share requests to cluster100 with the following hosts: member1, member2 and member3. In this case, the cluster alias is cluster100.</td>
</tr>
<tr>
<td>Coordinator address</td>
<td>The address of the machine to make the coordinator. If you started the creation process from a machine in a non-default state</td>
</tr>
<tr>
<td>Fault tolerance</td>
<td>Select the checkbox to enable fault tolerance. Fault tolerance enables the cluster to continue to operate even when some machines have failed. It works by ensuring that all data is stored on multiple machines. However, total storage capacity and overall performance are reduced as a consequence.</td>
</tr>
</tbody>
</table>
| Member addresses   | Enter the IP address or hostname information for the new members to add to the cluster. It can be in one of the following formats:  
  - IPv4 address (for example 192.168.1.100). Labeled v4.  
### Field name | Details
--- | ---
| Hostname (for example wilkoapp2.tideway.com). Labeled Hn. The addresses or hostnames used must resolve between all cluster members. |

4. **Click Create** to create the cluster.  
   The operation is added to the Pending Changes section.

5. Additionally a banner is displayed stating what the pending cluster changes are.

![Shows the Cluster Management page where a two machine cluster is going to be created.](image)

6. **Click Commit Changes** to create the cluster.  
The services on the machines shut down, the cluster changes are made, and the services restarted. The **Cluster Management** page is refreshed showing the progress of the cluster creation.

![Shows the cluster management page.](image)

7. When the changes are complete, and the machines have restarted, you need to log in again. The Cluster Management page shows the completed cluster. In this example, the cluster is rebalancing in the background, but is fully usable.
Adding a machine to an existing cluster

Once you have created a cluster, you can access the **Cluster Management** page and add or remove additional machines to or from the cluster. You can only add machines to the cluster if they are in the default configuration. If an appliance has *any* changes from the default configuration, you can create a cluster with it as the coordinator, but you cannot add it to an existing cluster without resetting its configuration.

To add a machine to an existing cluster

1. On the **Administration** tab, click **Cluster Management** in the Appliance section.

2. On the **Cluster Management** page, click **Add Members**.

   The **Add Members** dialog displays.

3. Enter the IP address or hostname information for the new members to add to the cluster in the **Member Addresses** field. It can be in one of the following formats:
3. IPv4 address (for example 192.168.1.100). Labeled v4.

4. Click Add to add the new member to the cluster. This becomes a pending change which needs to be committed before the member joins the cluster.

![Cluster Management Page with a New Member Waiting to Be Added](image1)

Shows the cluster management page with a new member waiting to be added. The dialog uses pills.

5. Click Commit Changes to add the new member to the cluster.

![Cluster Management Page with a New Member Being Added and That Activity Shown as Joining](image2)

The Cluster Management page is refreshed showing the progress of the member addition.

Shows the cluster management page with a new member being added and that activity shown as joining.

The state of the new member is shown as joining. You may see transient errors during this process, for example, the coordinator may attempt to contact a service that has not yet started. These transient errors will clear and can be ignored. If you attempt to log into a BMC Atrium Discovery machine that is joining, an offline page is shown containing progress messages about the joining process.

**To reset the configuration of a machine**

1. On the Administration tab, click Cluster Management in the Appliance section. Where the machine is in a non-default configuration, the configuration changes are listed in the Summary pane.
Shows the cluster manager window for a non-default machine. A Reset Configuration button is provided.

2. Click **Reset Configuration** to reset the configuration of the machine to defaults.

**Warning**

Resetting the configuration of the machine destroys all data held in that machine.

The services are shut down and the machine displays the Reset Configuration screen.

Changing the machine that is the coordinator

During the operation of a cluster, you might need to change the machine that is the coordinator of the cluster. For example, you may need to:

- remove the machine currently acting as the coordinator
- shut down the existing coordinator for maintenance
- troubleshoot problems with the coordinator
You can change the machine that is the coordinator using the **Cluster Management** page. You can make a member the coordinator by selecting **Make coordinator** from the Actions menu corresponding to the that machine in the Current Members list. You can also make another cluster member the coordinator by selecting **Make coordinator** from the Actions menu corresponding to the other machine. You can also perform other cluster management tasks from this page.

### Changing the machine that is the coordinator

To change the machine that is the coordinator:

1. On the **Administration** tab, click **Cluster Management** in the Appliance section.
2. On the **Cluster Management** page, select **Make coordinator** from the **Actions** list next to the cluster member you want to make the coordinator.
3. Confirm that you want to make the selected machine a coordinator

⚠️ **Note**

As a part of this operation all the services on all machines in the cluster are restarted.

An offline page is shown containing progress messages about the changing coordinator process. If you attempt to log into any machine that is a member of the cluster, the same offline page is shown.

### Changing the coordinator when the coordinator has failed

⚠️ This procedure can only be executed in a cluster with fault tolerance enabled.

To make a machine the coordinator when the existing coordinator has failed:

1. Access the UI of the machine you want to make the coordinator.
2. On the **Administration** tab, click **Cluster Management** in the Appliance section.
3. On the **Cluster Management** page select **Become Coordinator**. If the **Become Coordinator** button is not visible, reload the page in the browser.

4. Confirm that you want to make the selected machine the coordinator. This operation takes place immediately after you click OK.

   ![Image](image1.png)

   **Warning**

   This will forcibly remove all failed machines, including the current Coordinator, and restart Discovery services on all remaining machines in the cluster.

   An offline page is shown containing progress messages about the changing coordinator process. If you attempt to log into any machine that is a member of the cluster, the same offline page is shown.

   ![Image](image2.png)

   When the operation has completed, the **Cluster Management** page is refreshed and shows the previous coordinator in the Previous Members section.

   ![Image](image3.png)

   **Restoring a failed and forcibly removed coordinator**

   If a coordinator fails and is forcibly removed from the cluster, it must be reverted to a standalone state before it can be used again. To do this:
1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` to revert the machine to a standalone state. Enter:

```
[tideway@wilkoapp1 ~]$ tw_cluster_control --revert-to-standalone
Removing this machine from the current cluster

WARNING: This will delete all data on this cluster member!

Do you wish to proceed? (yes/no) yes
Stopping Cluster Manager service:
``` ...

The machine can be added back into the cluster.

Removing a machine from a cluster

Once you have created a cluster, you can access the Cluster Management page and add or remove machines to or from the cluster. Depending on the current status of the machine (healthy or unavailable), the remove operation runs in a normal or forced remove mode. Any healthy machine that you remove from a cluster is reset to the default configuration. Forced removal is used only when the communication with the machine has failed, for example, the machine is broken or permanently down, or the BMC Atrium Discovery services are not responding. When the machine is not responding and is removed forcibly, no changes are made to the machine configuration.

You can also perform other cluster management tasks from this page and from the command line utility.

⚠️ Before removing a machine that is currently a cluster coordinator, you should make another machine the coordinator.

To remove a machine from a cluster

1. On the Administration tab, click Cluster Management in the Appliance section.
2. On the Cluster Management page, either:
   a. Select the check box on the left hand side of the machine row in the Current Members list and select Remove from the Actions drop down.
   b. From the Actions drop down on the cluster machine row in the Current Members list and select Remove.

   The operation is added to the Pending Changes section and a banner is displayed stating what the pending cluster changes are.
For normal removal
If a healthy machine becomes unresponsive while the remove operation is pending, you get an alert about the status change and the remove operation fails when submitted. You can still force remove the machine by clicking **Force Removal?** option next to the alert.

Shows the pending changes with a broken machine awaiting to be removed normally.

For force removal
If an unresponsive machine becomes available while the remove operation is pending, you get an alert about the status change and the force removal fails when submitted. You can still remove the machine normally by clicking **Remove Normally?** option next to the alert.

Shows the pending changes with a machine awaiting to be forcefully removed.

3. Click **Commit Changes** to remove the machine from the cluster.
   The summary is updated to show the progress of the datastore rebalancing. When this is complete, the machine row is moved to the **Previous Members** section.

Shows the Cluster Management page where a machine is leaving.
If you used the removed machine's UI to commit the changes, an offline page is shown containing progress messages about the leaving process. If you attempt to log into a BMC Atrium Discovery machine in this state, an offline page is shown containing progress messages about the leaving process.
Shows the offline Cluster Management page progress messages about the leaving process.

A cluster is not removed, it stops being a cluster when the last member is removed from the coordinator.

**Changing the address of a machine**

When a cluster is in use, it may be necessary to change the address used to access a machine. For example, you may have used IP addresses to access machines, but would now like to use hostnames.

To change the address of a machine:

1. On the **Administration** tab, click **Cluster Management** in the Appliance section.
2. From the **Actions** drop down on the cluster machine row in the Current Members list, select **Change Address**.
   The **Change Member Address** dialog is displayed.
3. Enter the new IP address or hostname into the **Address** field using one of the following formats:
   - IPv4 address (for example **192.168.1.100**). Labeled **v4**.
   - IPv6 address (for example **2001:500:100:1187:203:baff:fe44:91a0**). Labeled **v6**.
   - Hostname (for example **wilkoapp3.tideway.com**). Labeled **Hn**.
4. If you are changing the address of the machine whose UI you are using, an additional **Show Known Addresses** link is shown. Click this to show all of the addresses that refer to this machine.
   a. Click on a displayed address to populate the **Address** field.
5. Click **Ok** to change the address. This becomes a pending change which needs to be committed before the address is changed.
6. Click **Commit Changes** to change the address.
   If the hostname or address entered does not refer to the correct machine then the changes are not applied.

### Reverting a cluster member into a standalone machine

#### Warning

This is a recovery operation. It might break the cluster and you might lose data. All data is removed from the machine that is reverted to a standalone machine. If the cluster does not have fault tolerance enabled, or if multiple machines have failed, you will lose all data in the cluster.

You might need to convert a member of the cluster into a standalone machine in the following circumstances:

- When the machine that was unresponsive and was forcibly removed from the cluster becomes available. If you cannot force remove the machine using the UI, you can do so from the command line. See the user example.
- When you cannot recover the whole cluster as most machines in the cluster are permanently down, but you want to reuse the machines.

Use the `tw_cluster_control` command line tool with the `--revert-to-standalone` option on the machine you need to make standalone.

### Removing all failed machines from the cluster

You can do this by running `tw_cluster_control` from any running member of the cluster. The order of these steps is particularly important. See the note below.

1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` to remove all failed machines from the cluster. You need to supply the password of the system user. Enter:

   ```
   [tideway@wilkoapp1 ~]$ tw_cluster_control --remove-broken
   Password:
   
   Found 1 broken member:
   Harness Cluster-03 (wilkoapp3.tideway.com)
   
   Are you sure you want to make these changes? (y/n) y
   
   1 member is being removed
   [tideway@wilkoapp1 ~]$ 
   ```
3. Before you can reuse any machine removed from the cluster, you must revert it to a standalone configuration. All data is lost in this process. On the command line of the machine you want to revert, enter:

```
tideway@wilkoapp3 ~> $ tw_cluster_control --revert-to-standalone
Removing this machine from the current cluster

WARNING: This will delete all data on this cluster member!

Do you wish to proceed? (yes/no) yes
Stopping Cluster Manager service:
Stopping Cluster Manager service: [ FAILED ]
Removing cluster configuration
Starting Cluster Manager service:
Starting Cluster Manager service: [ OK ]
Removing appliance configuration
Deleting datastore
Starting Security service: [ OK ]
...  
Starting Reasoning service: [ OK ]
Importing TKU packages: [ Unknown ]
Writing TKU-Core-2064-02-1-ADDM-10.0+: [ OK ]
[ TKU-Extended-DB-Discovery-2064-02-1-ADDM-10.0+: [ Unknown ]
[ TKU-Extended-Middleware-Discovery-2064-02-1-ADDM-10.0+: [ Unknown ]
[ TKU-System-2064-02-1-ADDM-10.0+: [ Unknown ]
Activating patterns: [ OK ]
Starting Tomcat: [ OK ]
Starting Reports service: [ OK ]
Starting Application Server service: [ OK ]
Updating cron tables: [ OK ]
Updating baseline: [ OK ]
Finished removing cluster configuration
[tideway@wilkoapp3 ~> $]
```

⚠️ If a machine in the cluster has failed, and you `--revert-to-standalone` before removing it from the cluster, then you must shut down the reverted machine before issuing the `--remove-broken` command.
Shutting down and restarting machines in a cluster

For some maintenance tasks you may need to restart the services on a machine, and for others you need to reboot or shut down a machine. For example, you need to:

- Restart services
- Reboot
- Shut down

For all of these options on individual machines, the cluster must have fault tolerance enabled.

**Warning**

The control operations on the Cluster Management page apply to *every* machine in the cluster. Individual machines are controlled from the **Actions** drop down.

To restart services on a machine in a cluster

You must be logged in as a user who is a member of the system group to restart the services on a machine in a cluster.

1. On the Cluster Management page, select **Restart Cluster Services** from the Actions drop down.
2. Confirm the action.

To reboot a machine in a cluster

You must be logged in as a user who is a member of the system group to reboot a machine in a cluster.

1. On the Cluster Management page, click **Reboot** from the Actions drop down.
2. Confirm the action.

To shut down a machine in a cluster

You must be logged in as a user who is a member of the system group to shut down a machine in a cluster.

1. On the Cluster Management page, select **Shutdown** from the Actions drop down.
2. Confirm the action.
Shutting down and restarting clusters

For some cluster maintenance tasks you may need to restart the services on all machines in the cluster, for others you need to reboot or shut down all machines in the cluster, and for some you can place the cluster into maintenance mode. For example, you need to:

- Restart services — after making changes to some discovery configuration options
- Shut down — to make hardware changes such as adding disks
- Reboot — after upgrading the BMC Atrium Discovery application software or upgrading the operating system

This functionality is also available in the Appliance Control page.

Maintenance mode is a user mode in which the only users permitted are those who are members of the system group. All users who are not members of this group are logged off the cluster and an explanatory message is displayed.

⚠️ Note

The control buttons at the top of the Cluster Management page apply to every machine in the cluster. Individual machines can be controlled from the Actions drop down page.

To shut down all machines in the cluster

You must be logged in as a user who is a member of the system group to shut down all machines in the cluster.

2. Confirm the action.

To reboot all machines in the cluster

You must be logged in as a user who is a member of the system group to reboot all of the machines in the cluster.

2. Confirm the action.

To restart services on all machines in the cluster

You must be logged in as a user who is a member of the system group to restart the services on all machines in the cluster.

2. Confirm the action.
To put the cluster into maintenance mode

You must be logged in as a user who is a member of the system group to place the cluster into maintenance mode.

2. Confirm the action.

All users who are not members of this group are logged off. System group users' screens are refreshed and the Quit Maintenance Mode button is displayed.

Maintenance mode is not a single-user mode. If you are performing any tasks that could affect other users (such as appliance backup) you should ensure that you are the only user logged in. Use the Administration => Security => Active Sessions window to verify this.

When non-system users are logged out, the login banner is displayed with an "under maintenance" message. When logging into an appliance that is in maintenance mode, you should ensure that your work does not interfere with that of other logged in users.

In maintenance mode, a flashing banner is displayed at the top of all pages. The flashing banner is a link to the Control page.

To take the cluster out of maintenance mode

To leave maintenance mode, click Quit Maintenance Mode.

Consolidation

Consolidation refers to the centralization of discovery data from scheduled or snapshot scans on multiple scanning appliances to one or more consolidation appliances. You might want to use consolidation in the following scenarios:

- **Firewalled environments**: When an environment is divided by firewalls so that a single appliance is unable to reach all parts of the network, a scanner can be situated on each section of the network blocked by a firewall. The scanners can all feed back data to a central consolidator.
- **Restricted (policy) networks**: Certain lines of business might enforce policies on the control of IT infrastructure in their environments. Where such policies limit or prohibit access, scanners can be deployed which all feed back data to a central consolidator.
- **Restricted (time) scanning windows**: Where a discovery window is short, a single appliance might be unable to complete a scan of a large range of IP addresses during the permitted time. Sharing the IP addresses between multiple scanners means each smaller scan can be completed in less time, and the results can be consolidated and viewed on the consolidator. You may consider using a cluster in this situation.
In each of these situations, multiple scanners can be deployed, and their data consolidated into a central consolidator. The consolidator is then used for reporting and provides a coherent view of the entire scanned network. A consolidator must be set as one which accepts connections or feeds from scanners. Scanners must in turn register with a consolidator.

**Consolidators can also scan**

Any consolidation appliance can also be used to perform discovery in its own right.

**IP address ranges**

Although consolidation can be used to scan a firewalled environment, it is essential that the IP address ranges scanned by each scanner belong to the same IP address space. That is, if two scanners scan the same address, they must both reach the same device. If the IP address spaces are not consistent across all the scanners, information on the consolidator can be missing or incomplete.

This restriction only applies to the addresses *scanned* by the scanners – if discovery targets possess other IP addresses, there is no need for *them* to belong to a consistent IP address space.

**Consolidator** — The main purpose of the consolidator is to report on data consolidated from a number of other scanners. A consolidator can also be used to perform discovery in its own right.

**Scanner** — The scanner appliance also operates as a *normal* appliance. The only difference is that it constantly sends discovery data to the consolidator. After setting up, this process is transparent to the user. A scanner must request and be approved on a consolidator appliance before it can send any data to the consolidator. This is described in Approving or rejecting a scanner request. A scanner can send consolidation data to more than one consolidator.

On the consolidator user interface, the **Currently Processing Runs** tab shows any local scans and any consolidation runs in progress. The **Currently Processing Runs** is described in The Discovery Status page. The tab is shown below:

This screen illustrates the discovery status page for a consolidator.
Restrictions

BMC Atrium Discovery version 9.0 introduced major changes in the data model. Consequently there is no support for consolidation to versions 9.0 and later from pre-9.0 appliances.

The consolidator’s service pack release must be the same or greater than the scanner. This is checked when you test the scanner-consolidator connection and when the scanner periodically checks that the consolidator is still accessible.

- A 9.0 consolidator can only accept data from a 9.0 scanner
- A 10.0 consolidator can accept data from a 9.0 and 10.0 scanner
- A 10.1 consolidator can accept data from a 9.0, 10.0, and 10.1 scanner
- A 10.2 consolidator can accept data from a 9.0, 10.0, 10.1, and 10.2 scanner

If you try to consolidate to an earlier version, warning messages are shown in the UI.

What is consolidated?

The consolidated data is the BMC Atrium Discovery Directly Discovered Data (DDD) nodes including the data collected by the patterns. The data inferred by the scanners, for example, Software Instance nodes, is not consolidated, but the consolidator will infer it again (based on its pattern configuration).

⚠️ TKU release, patterns, CSV imports and consolidation

The TKU release package and custom patterns that are loaded on the scanning and consolidators must be the same in order to infer the same data, for example, Software Instance nodes. This is not enforced in any way by the system.

The data imported via CSV in a scanner will not be consolidated. It has to be imported into all other appliances too.

Missing information when patterns run commands on other hosts

When a host is discovered and patterns are triggered which run commands on a second host, the DDD on both hosts is updated. When the original host is consolidated, the DDD on the second host is not available to the patterns that trigger on the consolidator. When the second host is consolidated, the DDD created on it when discovering the first host is not included. Consequently the consolidator will always report that the information from the second host is unavailable. The error "Request for information not part of the consolidated data" will be reported in the consolidated DiscoveryAccess. This can lead to missing nodes (licensing Detail, SoftwareComponents, and so on) and relationships on the consolidator. To work around this behavior, scan the original host from the consolidator.
Configuring consolidation

Configuring consolidation is a two step procedure. Initially the appliance which is to be the consolidator must be set as a consolidator, and then one or more scanners register with the appliance. To configure consolidation you need the permissions detailed in Consolidation Permissions.

Consolidation and clusters

When using standalone scanners to consolidate to a cluster, use the IP address of the coordinator (displayed in the cluster UI) as the consolidation target. If you subsequently change the coordinator of the target cluster, you must update the consolidation configuration on the scanner.

When using a cluster as a scanner, you can configure consolidation using any member UI, but only the coordinator of the cluster sends information to the consolidator. The coordinator of the target cluster must be specified as the consolidation target.

⚠️ Firewalls and consolidation

Consolidation uses port 25032 to communicate. The scanner must be able to connect to port 25032 on the consolidator. You must configure any firewalls between scanners and consolidators to allow this traffic. For clusters that act as scanners you must open port 25032 on all members. For clusters that act as consolidators you must open port 25032 on the coordinator, but if you change the coordinator you must open port 25032 on the new coordinator.

To set an appliance as a consolidator

1. From the Discovery section of the Administration tab, select Discovery Consolidation. The Consolidation page is displayed.
   You cannot use consolidation if the appliance is named Discovery_Appliance. A warning is displayed including a link to where you can change the appliance name.
2. In the Consolidation page, click Set as Consolidation Appliance.
   The appliance is now configured as a consolidator.

To set an appliance as a scanner

1. From the Discovery section of the Administration tab, select Discovery Consolidation.
2. In the Consolidation page, click Set as Scanning Appliance.
   This dialog enables you to specify a Consolidation target. Enter or edit the following information in the dialog:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Details</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Field Name</td>
<td>The name of the scanner. Names must be unique in the consolidation network and you cannot consolidate a scanner with the default name, Discovery_Appliance. The name is taken from the Administration =&gt; Appliance Configuration =&gt; Identification page. See Initial configuration. A change link is provided which displays the Identification page. In the identification page you can change the name of the appliance. You can only consolidate appliances which have unique names.</td>
</tr>
</tbody>
</table>
| Consolidation Appliance | The address of the consolidator. This can be specified as one of the following:  
• Hostname or FQDN  
• IPv4 or IPv6 address  

⚠️ You can supply credentials for the consolidation appliance in this dialog. If you supply valid credentials here, the scanner is approved automatically. |
| Username         | The user name for a user on the consolidator. This user must have appropriate permissions to approve the connection of the scanner to the consolidator. |
| Password         | The password for the user on the consolidator.                                                                                           |

If the target consolidator is an earlier version that the scanner, you are warned that the consolidator version is too old.

If you supplied valid credentials for automatic approval on the consolidator, then the scanner is now configured.

**To add an additional consolidator**

A scanner can send consolidation data to more than one consolidator. To do so:

1. Click the **Add new Consolidation Appliance** button.
   
The Add New Consolidation Appliance dialog is displayed. This is described above.

2. Enter the details of the consolidator and, if required, the username and password for automatic approval.

3. Click submit to apply the changes.

**Approving or rejecting a scanner request**

After a request (without automatic approval) has been made from a scanner, it requires approval on the consolidator.

To approve or reject a pending scanner request:

1. From the **Administration** tab on the consolidator, select **Discovery Consolidation** from the Discovery section.

   In the following example, the "de-32.tideway.com" appliance has requested to become a scanner.
This screen shows that the de-32.tideway.com appliance has requested to become a scanner.

- To accept the appliance connection, click **Approve**.
- To reject the request, click **Reject**. When you do this, the connection is deleted from the consolidator and when no connections remain the scanner reverts back to a non-consolidated machine.

**When consolidation is running**

Once consolidation has been set up, whatever scanning takes place on the scanner is automatically sent to the consolidator as soon as possible after the scan of an endpoint is complete. On the consolidator, runs are displayed that are marked specifically as consolidation runs and can be viewed from the **Discovery Status page**.

Discovery must be running on the consolidator for consolidation to take place. If Discovery is not running, the consolidator will refuse to accept data from the scanner. The scanner will attempt to resend data later. Also, if Discovery is stopped on the consolidator, it will stop consolidating any data it has already received.

**Canceling consolidating discovery runs**

You can cancel a consolidating discovery run from the scanner or from the consolidator. Where possible you should always cancel the discovery run on the scanner. This is done by selecting the discovery run on the **Discovery Status page** of the scanner and clicking **Cancel Runs**.

Canceling the discovery run at the scanner enables the consolidator to finish receiving data from the scanner. This stops the scan rather than the consolidation so that the two appliances' data remains consistent.

⚠️ **Canceling a consolidation Run on the consolidator stops the consolidation though the scan continues on the scanner. This leads to inconsistencies between the data on the two appliances. Where possible you should always stop the scan on the scanner and allow the consolidation to run to completion.**

If you must cancel a consolidation run from the consolidator, you can do so by selecting the discovery run on the Discovery Status page of the consolidator and clicking **Cancel Runs**. If there are problems canceling the consolidation run, a status message is displayed.
Logging in to the appliance command line

Certain features of BMC Atrium Discovery need command line access to configure. For example, the appliance obtains its IP address using DHCP by default, though you can configure it at the command line to use a static IP address.

Default command line passwords

BMC Atrium Discovery is configured with the following default command line user accounts and passwords.

- root — tidewayroot
- tideway — tidewayuser
- netadmin — no password configured.
- upload — no password configured. You cannot log in as the upload user, it is only intended for loading scanner files onto the appliance. See the upload user for more information.

The first time you log onto the appliance UI, you are forced to change the UI system user password and the passwords for the tideway and root command line users. To change the command line user passwords subsequently, log in using the VMware Server Console, or ssh in to the appliance using a terminal. You cannot ssh into the appliance as the root user. First you must log in as the tideway user and su to root.

Updated SSH server

The SSH server installed on the BMC Atrium Discovery 10.2 appliance has been modified. This modification removes some weak ciphers and HMAC algorithms (and non-FIPS approved) from the list of allowed connections. In practice this should not affect most users, however, some older versions of ssh clients will be unable to connect.

The following versions, or later, of the following clients have been tested and are known to work:

- PuTTY (0.64)
- WinSCP (5.7.2)
- mRemoteNG (1.72 using ssh2)
- MobaXterm (6.5)
- KiTTY (0.63.2.2)
- Bitvise SSH Client (6.22)

Password quality

The default requirement for command line passwords is that they must have at least one lowercase letter, one uppercase letter, one numeric character, and one special character. They must also contain a minimum of six characters, at least 4 characters must have been changed against the last password, and there must be no more than three repeated characters.
To change the root and tideway passwords

1. As the root user, use the `passwd` command to change the password.
2. Enter a new password when prompted. An example session is shown below:

   ```
   # passwd
   Changing password for user root.
   New UNIX password:
   Retype new UNIX password:
   passwd: all authentication tokens updated successfully.
   #
   ``

When you enter the new password it is not echoed to the screen.

1. Change the tideway user’s password. Still logged in as the root user, change the password using the `passwd` command specifying the tideway user.
2. Enter a new password when prompted. An example session is shown below:

   ```
   # passwd tideway
   Changing password for user tideway.
   New UNIX password:
   Retype new UNIX password:
   passwd: all authentication tokens updated successfully.
   #
   ``

   The passwords are changed.

The root user

The root user has unlimited privileges on the appliance and should only be used for the following tasks:

- Changing the default Linux passwords
- Upgrading the BMC Atrium Discovery application software
- Adding new disks to the appliance

Any other task on the appliance which typically would require a UNIX/Linux root user can be undertaken using `sudo` to temporarily grant additional privileges to the tideway user.

The tideway user

The tideway user is a powerful administrative account intended for performing virtually all of the tasks that would need to undertake using the command line. Examples of such tasks are:

- All command line utilities
Customizations (taxonomy, reports, and visualizations)
- Managing log file (though this can be done through the UI)
- Starting and stopping services (with additional privileges accessed through `sudo`)

### Additional users

The following additional users are configured on the appliance:

- **The netadmin user**
- **The upload user**

### The netadmin user

The netadmin user account is a special account with limited privileges. It enables you to change network, hostname and gateway settings without requiring root privileges. You can also use the netadmin user to reboot the appliance. If no password has been configured for the netadmin user, you can log in by entering:

```
su - netadmin
```

When you log in to BMC Atrium Discovery as the netadmin user, you see the following text based interface:

```
login as: netadmin
netadmin@ee-64's password:
Last login: Thu May 26 12:56:46 2011 from 192.168.0.1

Tideway Appliance Network Administration Shell
----------------------------------------------
G Configure General Settings
I Configure Network Interfaces
R Exit & Reboot the appliance
Q Exit *without* rebooting

Select option:
```

Select an option by entering the code letter:

- Enter **G** to configure the appliance hostname and the gateway details.
- Enter **I** to configure the appliance network interfaces.
- Enter **R** to exit the netadmin session and reboot the appliance.
- Enter **Q** to exit the netadmin session without rebooting the appliance.

⚠️ If you make any configuration changes from the General Settings or Network Interfaces menu, you must reboot the appliance for the changes to take effect.
The following tables describe the items in the General Settings and Network Interface menus:

**G - General Settings menu**

<table>
<thead>
<tr>
<th>Key</th>
<th>Label</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>Change</td>
<td>Enter H to set the hostname or FQDN. At the <strong>Hostname</strong> or <strong>FQDN</strong> prompt enter the new hostname or FQDN. Press enter to return to the General Settings menu.</td>
</tr>
<tr>
<td>G</td>
<td>Change</td>
<td>Enter G to change the gateway. At the <strong>Gateway</strong> prompt enter the gateway IP address. Press return and at the <strong>Gateway Device</strong> prompt enter the gateway device type (currently only &quot;eth0&quot; is supported).</td>
</tr>
<tr>
<td>D</td>
<td>Discard changes</td>
<td>Discard any changes and remain in the General Settings menu. Requires confirmation.</td>
</tr>
<tr>
<td>C</td>
<td>Commit changes</td>
<td>Commit any changes and remain in the General Settings menu. Requires confirmation.</td>
</tr>
<tr>
<td>Q</td>
<td>Return to the main menu</td>
<td>Discard any unsaved changes and return to the main menu. Requires confirmation.</td>
</tr>
</tbody>
</table>

**I - Network Interfaces menu**

<table>
<thead>
<tr>
<th>Key</th>
<th>Label</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reconfigure</td>
<td>Enter 1 to reconfigure eth0. At the <strong>Use DHCP</strong> prompt enter Y to use DHCP or N to use a static IP address. If you use a static IP address enter the IP address and Netmask when prompted. For DHCP and static IP addresses, choose whether to enable the setting at boot. At the <strong>Enable on boot</strong> prompt, enter Y or N.</td>
</tr>
<tr>
<td>A</td>
<td>Add interface</td>
<td>Enter A to add a new interface. At the <strong>Device</strong> prompt enter the new device name. For example eth1. At the <strong>Use DHCP</strong> prompt enter Y to use DHCP or N to use a static IP address. If you use a static IP address enter the IP address and Netmask when prompted. For DHCP and static IP addresses, choose whether to enable the setting at boot. At the <strong>Enable on boot</strong> prompt, enter Y or N.</td>
</tr>
<tr>
<td>R</td>
<td>Remove / Restore Network Interfaces</td>
<td>Enter R to list configured network interfaces. To remove or restore a listed interface, enter the menu number corresponding to the interface. A [DELETED] flag marks removed interfaces.</td>
</tr>
<tr>
<td>D</td>
<td>Discard changes</td>
<td>Discard any changes and remain in the Network Interfaces menu. Requires confirmation.</td>
</tr>
<tr>
<td>C</td>
<td>Commit changes</td>
<td>Commit any changes and remain in the Network Interfaces menu. Requires confirmation.</td>
</tr>
<tr>
<td>Q</td>
<td>Return to the main menu</td>
<td>Discard any unsaved changes and return to the main menu. Requires confirmation.</td>
</tr>
</tbody>
</table>

**R - Reboot the appliance**

Enter R to reboot the appliance. This operation requires confirmation, after which your session ends and the appliance reboots.
Q - Quit
Enter Q to quit the netadmin session.

Changing the appliance hostname

⚠️ **The netadmin user is the preferred way of configuring networking**

The preferred way of administering any of the appliance network configuration is to use the netadmin user account. The netadmin user account enables you to change network, hostname and gateway settings without requiring root privileges. You can also use the netadmin user to reboot the appliance.

This section describes how to set the appliance hostname and to ensure that it is locally resolved even if the IP address of the appliance changes.

You can set the hostname either locally, or using DHCP/DNS.

**To set the hostname locally**

If you want the appliance to use the same hostname for all NICs, set the hostname in the `/etc/sysconfig/network` file.

If you want to use different hostnames for individual NICs, set the hostname in the `/etc/sysconfig/network-scripts/ifcfg-ethX` file. X is the number of the interface, for example `eth0`.

Add the following line to the file:

```
HOSTNAME= london01
```

where `london01` is the new hostname.

If you do not have a DNS entry for the host, or you require a failsafe when DNS is unavailable, enter the hostname and FQDN in the `/etc/hosts` file. Use the appliance IP address if it is unlikely to change, or a loopback IP address other than `127.0.0.1` (which can interfere with the resolution of `localhost` and `localdomain`.

The following example uses `london01` as the hostname, `london.com` as the domain name, and 192.168.0.100 as the IP address. Add only one of the example entries.

```
127.0.0.1       localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
```
The following example uses the same host and domain names as above and shows commands to enter to ensure that the hostname is set correctly:

```
[tideway@london01 ~]$ hostname
london01
[tideway@london01 ~]$ hostname --domain
london.com
[tideway@london01 ~]$ hostname --fqdn
london01.london.com
[tideway@london01 ~]$ ping `hostname`
PING london01 (192.168.0.100) 56(84) bytes of data.
64 bytes from london01 (192.168.0.100): icmp_seq=1 ttl=64 time=0.050 ms
64 bytes from london01 (192.168.0.100): icmp_seq=2 ttl=64 time=0.047 ms
64 bytes from london01 (192.168.0.100): icmp_seq=3 ttl=64 time=0.042 ms
64 bytes from london01 (192.168.0.100): icmp_seq=4 ttl=64 time=0.041 ms

--- london01 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 0.041/0.045/0.050/0.003 ms
[tideway@london01 ~]$ 
```

To set the hostname using DHCP/DNS

Some networks are configured so that you can set your own hostname and update the DHCP and DNS servers accordingly.

To enable this for all NICs, set the `DHCP_HOSTNAME` in the `/etc/sysconfig/network` file.
To enable this for individual NICs, set the `DHCP_HOSTNAME` in the `/etc/sysconfig/network-scripts/ifcfg-ethX` file.
Where, X is the number of the interface. For example, eth0.

The following example uses `london01` as the hostname and `london.com` as the domain name.

```
DHCP_HOSTNAME=london01.london.com
```

When the appliance renews its IP address with the DHCP server, the DHCP server will also update the DNS server with the hostname and IP address of the appliance.
Diagnosing hostname problems

Some issues are indicative of incorrectly configured hostnames.

Visualizations and export do not display

This problem can be identified if you see a 500 Error - Internal Server Error! page in the user interface (UI) when attempting to load visualizations. You will also see the following error message in $TIDEWAY/tomcat/logs/catalina.out:

```
java.net.UnknownHostException: local_hostname: local_hostname
```

where, local_hostname is the hostname set on the computer.

To resolve this problem, see the previous section, Setting the hostname locally.

Cannot access the UI — 500 internal server error

This error can have many causes, as it results from one or more of the tideway services failing to start. A common cause however is that localhost cannot be resolved. If this is the case errors of the following form are displayed when restarting the tideway services.

```
[current date/time] [error] couldn't resolve WKServer address
[current date/time] [error] Couldn't resolve hostname for WebKit Server
```

Changing the appliance IP address

⚠️ The netadmin user is the preferred way of configuring networking

The preferred way of administering any of the appliance network configuration is to use the netadmin user account. The netadmin user account enables you to change network, hostname and gateway settings without requiring root privileges. You can also use the netadmin user to reboot the appliance.

Setting a static IP address for the first interface (eth0) requires the following steps:

1. Obtain networking information.
2. Stop the tideway services
3. Edit the network configuration files.
4. Restart networking.
5. Restart the tideway services.
Obtaining networking information

You must obtain the following information before starting:

- IP Address: in this example 192.168.0.100
- Netmask: in this example 255.255.255.0
- Gateway: in this example 192.168.0.1

Consult your system administrator for this information.

Stopping the Tideway services

Before changing any networking configuration, you must stop the following services:

- tideway
- cluster
- omniNames
- appliance

To do this, as the tideway user, enter the following:

```
[tideway@london01 ~]$ sudo /sbin/service tideway stop
[tideway@london01 ~]$ sudo /sbin/service cluster stop
[tideway@london01 ~]$ sudo /sbin/service omniNames stop
[tideway@london01 ~]$ sudo /sbin/service appliance stop
```

Editing the network configuration files

Edit Network Configuration Files Using the Console

Ensure you are on the console or have access to the console before making any networking changes.

The appliance is configured by default to use DHCP. To configure it to use a static IP address, you must edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file as root. In addition to the network settings obtained previously, you need to determine the MAC address of the network card.

To do this:

1. Enter the following command:
The MAC address is given after the heading `HWaddr`. In this example, the MAC address is `00:0C:29:BB:34:6D`.

The following example shows the default `ifcfg-eth0` file:

```
DEVICE=eth0
BOOTPROTO=dhcp
DHCPCLASS=
ONBOOT=yes
```

2. **Edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` and add the values obtained above.** The file will look like this:

```
DEVICE=eth0
BOOTPROTO=static
HWADDR=00:0C:29:BB:34:6D
TYPE=Ethernet
ONBOOT=yes
IPADDR=192.168.0.100
NETMASK=255.255.255.0
```

3. **Add the gateway address to the `/etc/sysconfig/network` file.**

You can edit the appliance hostname, but you must ensure that it still resolves correctly. See Changing the appliance hostname for more information.

The file will look like this:

```
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=london01
GATEWAY=192.168.0.1
```

**Restarting networking**

You must restart networking to make sure that the changes have been applied correctly. To do this, enter:
When networking starts, enter the following commands to ensure that the new networking information is showing correctly and that the appliance can resolve its own hostname correctly:

```
[sudo@london01 ~]$ sudo /sbin/service network restart
```

```
When networking starts, enter the following commands to ensure that the new networking information is showing correctly and that the appliance can resolve its own hostname correctly:

```
[sudo@london01 ~]$ /sbin/ifconfig eth0
eth0      Link encap:Ethernet HWaddr 00:0C:29:BB:34:6D
          inet addr:192.168.0.100 Bcast:192.168.0.255 Mask:255.255.255.0
...  
[sudo@london01 ~]$ ping `hostname`
PING london01 (192.168.0.100) 56(84) bytes of data.
64 bytes from london01 (192.168.0.100): icmp_seq=1 ttl=64 time=0.050 ms
64 bytes from london01 (192.168.0.100): icmp_seq=2 ttl=64 time=0.047 ms
64 bytes from london01 (192.168.0.100): icmp_seq=3 ttl=64 time=0.042 ms
64 bytes from london01 (192.168.0.100): icmp_seq=4 ttl=64 time=0.041 ms
--- london01 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 0.041/0.045/0.050/0.003 ms
```

If the appliance cannot resolve its own hostname (there is no response from the ping command), see Changing the appliance hostname for information on changing the hostname.

**Restarting the Tideway services**

The final test is to log in to the appliance user interface (UI) using a web browser. You can only do this after you have restarted the tideway services. Enter the following commands:

```
[sudo@london01 ~]$ sudo /sbin/service appliance start
[sudo@london01 ~]$ sudo /sbin/service omniNames start
[sudo@london01 ~]$ sudo /sbin/service cluster start
[sudo@london01 ~]$ sudo /sbin/service tideway start
```

Next, attempt to log in to the appliance UI using a web browser.

**Troubleshooting**

If you cannot log in through a web browser, try stopping and restarting the services:

```
[sudo@london01 ~]$ sudo /sbin/service tideway stop
[sudo@london01 ~]$ sudo /sbin/service cluster stop
[sudo@london01 ~]$ sudo /sbin/service omniNames stop
```
Try to ping the gateway or use traceroute to the gateway. In this example, 192.168.0.1 is the gateway address provided by your system administrator. Enter the following:

```
[tideway@london01 ~]$ ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=127 time=0.324 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=127 time=0.450 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=127 time=0.342 ms
64 bytes from 192.168.0.1: icmp_seq=4 ttl=127 time=0.401 ms
--- 192.168.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.324/0.379/0.450/0.051 ms
```

```
[tideway@london01 ~]$ traceroute 192.168.0.1
traceroute to 192.168.0.1 (192.168.0.1), 30 hops max, 40 byte packets
  1  192.168.1.4 (192.168.1.4)  4.225 ms  4.377 ms  4.532 ms
  2  192.168.0.1 (192.168.0.1)  0.265 ms  0.262 ms  0.252 ms
```

If these tests fail, contact your system administrator.

The upload user

The upload user account is a special account with limited privileges. You cannot log in as the upload user, it is only intended for loading scanner files onto the appliance. See **Standalone UNIX scanning** for a description of scanner files and their use.

When an appliance is built, the upload user has no password. Before you can use the upload user, you should log in as the root user and use the `passwd` command to change the password.

```
# passwd upload
Changing password for user upload.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
#`
```
Using screen

Screen is a utility which enables you to access multiple virtual terminals from a single terminal. For example, you can connect to the BMC Atrium Discovery appliance using a single terminal emulator, start screen, create multiple virtual terminals, and use one to start a long running process such as upgrading the appliance and another to tail the output log. Screen provides the additional advantage that should the connection be dropped, you can re-connect to the appliance, and re-attach to screen where the processes are still running, unaffected by the dropped connection.

Running screen

The following commands enable you to start screen and to reattach to an existing session. Screen offers many more command line options, but these are beyond the scope of this tutorial. See further information to find out more about screen.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>screen</td>
<td>Start screen.</td>
</tr>
<tr>
<td>screen -ls</td>
<td>List existing screen sessions. Used after detaching or after a dropped connection. Example output showing two screen sessions:</td>
</tr>
</tbody>
</table>

```
[tideway@appliance01 ~]$ screen -ls
There are screens on:
  2970.pts-0.appliance01 (Detached)
  3078.pts-0.appliance01 (Detached)
2 Sockets in /var/run/screen/S-tideway.
[tideway@appliance01 ~]$
```

| screen -r | Reattach to a detached screen session. If there are multiple screen sessions listed, add the session to which you want to connect. |

```
[tideway@appliance01 ~]$ screen -r 2970.pts-0.appliance01
```

Working with screen

When screen is running, you can run commands in the terminal as usual. To interact with screen you begin all commands with:

CTRL+a

That is, the **Control** key and the **a** key pressed at the same time. Then an additional key completes the command.

The following table shows some of the more common screen commands. For complete information on the commands available in screen, consult the man page.
1. Command: CTRL+a c
   Description: Create a new screen virtual terminal.

2. Command: CTRL+a n
   Description: Switch to the next virtual terminal.

3. Command: CTRL+a p
   Description: Switch to the previous virtual terminal.

4. Command: CTRL+a d
   Description: Detach the current terminal from the screen session. The screen utility continues to run and all virtual terminals and processes continue to run. You can reattach using the `screen -r` command described above.

5. Command: CTRL+a A
   Description: Rename the virtual terminal. An example is given below.

To close a virtual terminal, exit all running programs and then type exit to close the terminal.

**Exiting screen**

The following command exits the screen utility. This is not the same as detaching. When you exit screen, the utility stops. You can only do this when you have closed all your running programs.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL+D</td>
<td>Exit screen. You can also exit screen when you have closed all but one virtual terminal by typing exit in the remaining virtual terminal.</td>
</tr>
</tbody>
</table>

**Example screen usage**

The following example shows how you can use screen when upgrading the appliance to run the upgrade and monitor its progress. It also shows how to detach and reattach screen "windows" or virtual terminals.

This example procedure does not tell you how to upgrade the appliance, that is covered here.

**Using screen when upgrading the appliance**

1. Login to the appliance command line interface as the tideway user.
2. Run the screen utility. Enter:
   ```
   [tideway@appliance01 ~]$ screen
   ```
   You are presented with a welcome information screen. Press the Enter key to dismiss this.
3. Rename the virtual terminal to "twuser". Enter:
   ```
   CTRL+a A
   ```
   At the bottom of the terminal window highlighted text prompts you to Set window's title to:. Enter:
   ```
   Set window's title to: twuser
   ```
4. Create a new virtual terminal. Enter: 
   CTRL+a c

5. Become the root user. Enter the su command and the root password when prompted:

   [tideway@appliance01 ~]$ su
   Password:
   [root@appliance01 ~]#

6. Rename the virtual terminal to "twroot". Enter: 
   CTRL+a A
   At the bottom of the terminal window highlighted text prompts you to Set window's title to:. Enter:

   Set window's title to: twroot

7. Start the upgrade procedure. The following command shows running the script:

   [root@appliance01 ~]# sh ADDM_Upgrade_64_Vnnnnn_nnnnnn_dev.sh

   A welcome to the upgrade banner is displayed.

8. Once the script is running, switch to the first (twuser) virtual terminal. Enter: 
   CTRL+a p
   The tideway user prompt is displayed.

   [tideway@appliance01 ~]$

9. Upgrading the OS and the BMC Atrium Discovery application can take a long time. Monitor progress by checking the log file using the following command:

   [tideway@appliance01 ~]# tail -f/usr/tideway/log/upgrade_V.n.log

10. Switch back to the twroot virtual terminal. Enter: 
     CTRL+a n
     The upgrade is progressing:
11. Switch back to the twuser virtual terminal. Enter: 
CTRL+a n
The upgrade log is displayed:

```
2012-08-02 12:44:39: Move new IPv6 firewall configuration into place
2012-08-02 12:44:39: Ensure correct perms are set on the IPv6 firewall configuration
2012-08-02 12:44:39: Add appliance, omniNames and tideway services
2012-08-02 12:44:39: Set omniNames and httpd to run on runlevel 3
```

12. Someone from IT comes and tells you that they need to reboot your PC immediately. You then detach your terminal from the screen session before rebooting. Enter: 
CTRL+a d
13. Close the terminal window. Enter:

```
[tideway@appliance01 ~]$ exit
```

and reboot the PC.
14. Once the PC is running again, login to the appliance command line interface as the tideway user.
15. Reattach to the screen session. Enter:

```
[tideway@appliance01 ~]$ screen -r
```

Screen reconnects to the existing session and you can see how the upgrade is progressing in the twroot virtual terminal, and see the tailed log in the twuser virtual terminal. More information on reattaching is provided below.

The procedure above has demonstrated using screen with two named virtual terminals.

**Recovering from a lost connection using screen**

If you lose the connection to the appliance and you have used screen, you can reconnect to the appliance and recover the virtual terminal running the compaction. To do this:
1. Reconnect to the appliance and login as the tideway user.
2. List the current screen sessions. Enter:

```
[tideway@appliance01 ~]$ screen -ls
There is a screen on:
  23274.pts-0.appliance01        (Detached)
1 Socket in /var/run/screen/S-tideway.
```

If there is only one screen listed, you can re-attach to it with a simple command:

```
[tideway@appliance01 ~]$ screen -r
```

If there is more than one, copy the screen identifier:

```
[tideway@appliance01 ~]$ screen -r 23274.pts-0.appliance01
```

All of the virtual terminals are recovered and you can see how the upgrade is progressing.

**Further information**

This section lists some useful resources for screen.

- Screen website
- Screen documentation

**Using command line utilities**

BMC Atrium Discovery provides utilities that you can run on a command line interface to configure Discovery by obtaining information from specific systems. All the utilities are located in the /usr/tideway/bin directory and are named tw_utilityname.

**Supported utilities**

All command line utilities that are supported in BMC Atrium Discovery are documented in this section. Any utility that is not documented in this section is an explicitly unsupported tool.

- tw_scan_control
- tw_reasoningstatus
- `tw_model_init`
- `tw_passwd`
- `tw_upduser`
- `tw_check_reports_model`
- `tw_adduser`
- `tw_cron_update`
- `tw_disco_export_platforms`
- `tw_restore`
- `tw_backup`
- `tw_listusers`
- `tw_imp_csv`
- `tw_imp_ciscoworks`
- `tw_disco_import_platforms`
- `tw_baseline`
- `tw_tripwire_rebaseline`
- `tw_disco_control`
- `tw_excluderanges`
- `tw TERMINATE_WINPROXY`
- `tw_deluser`
- `tw_convert_reports`
- `tw_tax_export`
- `tw_sign_winproxy_config`
- `tw_query`
- `tw_config_dashboards`
- `tw_sync_control`
- `tw_tax_import`
- `tw_ds_offline_compact`
- `tw_tax_deprecated`
- `tw_top`
- `tw_pattern_management`
- `tw_atso_control`
- `tw_run_upgrade`
- `tw_model_wipe`
- `tw_cluster_control`
Duplicate or enhanced functionality in user interface

Many of the utilities perform functions that are already part the BMC Atrium Discovery user interface, and might duplicate what you might be able to accomplish more easily in the best-practice use of the product. Most of the configuration you perform will be done using the Discovery Configuration page accessed from the Administration tab; however, the functionality might be located in other areas of the product. Documentation for each utility in this section includes a link to the corresponding location in the user interface where you can perform the intended function.

⚠️ Use of command line utilities

Although command line utilities offer a potentially faster or more convenient way to perform a specific function, they might also cause unintended consequences that might compromise your environment if not used carefully.

When using command line utilities, particularly long-running processes such as `tw_ds_offline_compact` it is recommended that you use the screen utility to prevent any problems that might arise from dropped connections. An example of using screen is shown in the `tw_ds_offline_compact` documentation. Using screen provides more detailed information on screen and links to online resources.

Common options in the utilities

In each utility, you can specify options, or arguments, that enable you to specify what action you want the utility to perform. For many of the utilities, several options are inherited and their basic purpose does not change. In this section, common options are not described for each utility because they are not typically used.

The following table lists the common options that are inherited by many of the command line utilities available with BMC Atrium Discovery.

<table>
<thead>
<tr>
<th>Common Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h, --help</td>
<td>Displays the required use of the utility and describes the available options</td>
</tr>
<tr>
<td>--logbackupcount=NUM</td>
<td>Specifies the number of backup logs files that are preserved</td>
</tr>
<tr>
<td>--logfile=FILE</td>
<td>Specifies the log file that log messages are written to</td>
</tr>
<tr>
<td>--loglevel=LEVEL</td>
<td>Specifies the logging level as one of the following values:</td>
</tr>
<tr>
<td></td>
<td>* debug: logs all messages</td>
</tr>
<tr>
<td></td>
<td>* info: logs critical, error, warning, and information messages</td>
</tr>
<tr>
<td></td>
<td>* warn: logs critical, error, and warning messages</td>
</tr>
<tr>
<td>Common Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>• <code>error</code>: logs critical and error messages</td>
</tr>
<tr>
<td></td>
<td>• <code>crit</code>: logs only critical messages</td>
</tr>
</tbody>
</table>

```
--passwordfile=FILE
```

Specifies a file from which the password is to be read. This is only relevant for utilities that have the `--username` option. This file is not encrypted, though you can set the file permissions to owner-only (`chmod 600 passwordfile.txt`) to restrict access to the file. For more information about password policies, see Managing Security Policies.

```
-p, --password=PASSWD
```

Specifies the password of the BMC Atrium Discovery user. If no password is specified, you are prompted for one. This is only relevant for utilities that have the `--username` option. For more information about password policies, see Managing Security Policies.

```
-u, --username
```

The user to run the utility as. This has to be a valid BMC Atrium Discovery UI user such as the system user, not a username used to access the command line via ssh.

```
-v, --version
```

Displays the BMC Atrium Discovery version number, copyright information, and the name of the utility.

---

**Warning**

You should not use any service utilities (those named `tw_svc_name`), because the improper use of a service-based command could have potentially adverse results on your system.

**tw_scan_control**

The `tw_scan_control` command line utility enables you to do the following:

- perform immediate and scheduled scans of IP addresses or ranges
- start and stop Reasoning
- disable and enable scheduled scans
- remove and update scheduled scans

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_scan_control` command line utility (see Scanning IP addresses or ranges). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

This page contains the following sections:

- **Using the `tw_scan_control` utility** – This section contains the general guidelines to use the `tw_scan_control` utility.
- **Common options to manage snapshot and scheduled scans** – This section contains information about the common options to manage snapshot and scheduled scans, such as starting and stopping discovery and adding discovery runs.

- **Options to manage scheduled scans** – This section contains information about the options to manage scheduled scans, such as adding scheduled discovery runs, enabling and disabling scheduled discovery runs, listing scheduled discovery runs, and updating and deleting scheduled discovery runs.

- **Overlapping of scheduled scans and excludes** – This section contains information about the expected behavior in the event of overlapping of scheduled scans and excludes.

### Using the `tw_scan_control` utility

To use the utility, type the following command:

```bash
tw_scan_control [options] range
```

where:

- **range** is a single entry or a space separated list of IP address information in the following formats:
  - IPv4 address (for example `192.168.1.100`).
  - IPv4 range (for example `192.168.1.100-105, 192.168.1.100/24`, or `192.168.1.*`).
  - If you use the `--file` option, a range refers to a file that contains IP addresses.
- **options** are any of the options described in the following table and the common command line options described in [Using command line utilities](#).

⚠️ Scanning the following address types is not supported:
- IPv6 link local addresses (prefix `fe80::/64`)
- IPv6 multicast addresses (prefix `ff00::/8`)
- IPv6 network prefix (for example `fda8:7554:2721:a8b3::/64`)
- IPv4 multicast addresses (`224.0.0.0` to `239.255.255.255`)

In each of the sections below, user examples have been included for your reference. In these examples, the user name is system and the password is not specified on the command line. The utility prompts for the password after you enter the command. Type the commands on a single line; line breaks are provided in the examples to make them easier to read. Note the following:

- If you want to scan a range of IP addresses listed in a file, make sure that the IP addresses are arranged in a newline separated list.
With --enable, --disable, --remove, and --update=ID you are required to enter the list of corresponding range IDs. You can view the list of range IDs by using the --list and --list-full command line options.

Common options to manage snapshot and scheduled scans

⚠️ The common command line options are described in Using command line utilities.

By using the common options for snapshot and scheduled scans with the tw_scan_control command line utility, you can perform the following:

- Start and stop discovery
- Add discovery runs

Starting and stopping discovery

Use the following common options with the tw_scan_control command line utility to start and stop discovery:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-s, --start</td>
<td>Starts Reasoning. This is equivalent to clicking START ALL SCANS.</td>
</tr>
<tr>
<td>-x, --stop</td>
<td>Stops Reasoning. This is equivalent to clicking STOP ALL SCANS.</td>
</tr>
</tbody>
</table>

User examples:

To start discovery:

tw_scan_control --start

To stop discovery:

tw_scan_control --stop

Adding discovery runs

Use the following common options with the tw_scan_control command line utility to add snapshot and scheduled scans and specify run details:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a, --add</td>
<td>Adds a new scan range.</td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-r, --random</td>
<td>Scans the IP addresses (located in a file or listed at the command prompt) in random order.</td>
</tr>
<tr>
<td>-l, --scan-level=string</td>
<td>Specifies the scan level to use. This can be one of the following: Sweep Scan — This will do a sweep scan, trying to determine what is at each endpoint in the scan range. It will attempt to login to a device to determine the device type. Full Discovery — Retrieve all the default info for hosts, and complete full inference. This is the default if the option is not specified.</td>
</tr>
<tr>
<td>--company=companyname</td>
<td>Specifies the company name to use for a scan in a multitenant deployment.</td>
</tr>
<tr>
<td>-f, --file</td>
<td>Specifies a file or a list of files as arguments. They must be plain text files with a new line delimited list of IP addresses.</td>
</tr>
<tr>
<td>--label=label</td>
<td>Specifies the label for the scan.</td>
</tr>
<tr>
<td>--passphrase=passphrase</td>
<td>Specifies the vault passphrase to use.</td>
</tr>
<tr>
<td>--silent</td>
<td>Turns off informational messages.</td>
</tr>
</tbody>
</table>

User examples:

To specify an immediate scan of a single IP address:

```
tw_scan_control --add 192.168.0.1
```

To specify an immediate scan of a range of IP addresses:

```
tw_scan_control --add 192.168.0.1-10
```

To specify an immediate scan of IP addresses listed in a file

```
tw_scan_control --add --file ~/scanlist
```

For user examples related to scheduled scans, see the Options to manage scheduled scans section.

Options to manage scheduled scans

By using the scheduled scan options with the `tw_scan_control` command line utility, you can perform the following:

- Enable and disable scheduled discovery runs
- Add scheduled discovery runs
- List scheduled discovery runs
- Update and delete scheduled discovery runs

Enabling and disabling scheduled discovery runs

Use the following common options with the `tw_scan_control` command line utility to enable or disable scheduled scans:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--enable</td>
<td>Enables the chosen scheduled scans.</td>
</tr>
<tr>
<td>--disable</td>
<td>Disables the chosen scheduled scans.</td>
</tr>
</tbody>
</table>

User examples:

To disable a scheduled scan:

```
tw_scan_control --disable 6ee6e73209c64a4e9c0a0a8148a76f8b
```

To enable a scheduled scan which has been disabled:

```
tw_scan_control --enable 6ee6e73209c64a4e9c0a0a8148a76f8b
```

Adding scheduled discovery runs

You can schedule discovery runs which are daily, weekly, and monthly for which the following options with the `tw_scan_control` command line utility are available:

- Common options for scheduled discovery runs
- Options for scheduling daily discovery runs
- Options for scheduling weekly discovery runs
- Options for scheduling monthly discovery runs
- Backward compatible options

Common options for scheduled discovery runs

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--start-time=HH:MM</td>
<td>Sets the start time of a scheduled scan.</td>
</tr>
<tr>
<td>--end-time=HH:MM</td>
<td>Sets the end time of a scheduled scan.</td>
</tr>
<tr>
<td>--duration=DD:HH:MM</td>
<td>Sets the duration of scheduled scans.</td>
</tr>
<tr>
<td>--no-end-time</td>
<td>Sets the scheduled scan to run to its completion.</td>
</tr>
</tbody>
</table>

Options for scheduling daily discovery runs
Command Line Option | Description
--- | ---
--daily | Adds a daily scheduled scan.

User examples:

To specify a daily scheduled scan of a range of IP addresses with specified start and end time and label it TEST:

```
tw_scan_control --daily --start-time=21:30 --end-time=23:30 --label=TEST --add 192.168.0.1-10
```

To specify a daily scheduled scan of IP addresses listed in a file:

```
tw_scan_control --daily --start-time=21:30 --end-time=23:30 --add --file ~/scanlist
```

To specify a daily scheduled scan of a range of IP addresses with specified start time and duration:

```
tw_scan_control --daily --start-time=21:30 --duration=00:06:30 --add 192.168.0.1-10
```

Options for scheduling weekly discovery runs

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--weekly</td>
<td>Adds a weekly scheduled scan.</td>
</tr>
<tr>
<td>--weekly-start-week-days=WEEKDAYS</td>
<td>Sets the weekly scheduled scan start week day. The range of the start weekday is monday, tuesday, and so on.</td>
</tr>
<tr>
<td>--weekly-end-week-day=WEEKDAY</td>
<td>Sets the weekly scheduled scan end week day. The range of the end weekday is monday, tuesday, and so on.</td>
</tr>
</tbody>
</table>

User example:

To specify a weekly scheduled scan of a range of IP addresses with specified start time and day and specified end time and day:

```
tw_scan_control --weekly --weekly-start-week-days=monday --weekly-end-week-day=tuesday --start-time=21:30 --end-time=20:30 --add 192.168.0.1-10
```

Options for scheduling monthly discovery runs
<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--monthly</td>
<td>Adds a monthly scheduled scan.</td>
</tr>
<tr>
<td>--monthly-start-day=DAY</td>
<td>Sets the monthly scheduled scan start day. The range of the day is from 1 to 31.</td>
</tr>
<tr>
<td>--monthly-end-day=DAY</td>
<td>Sets the monthly scheduled scan end day. The range of the day is from 1 to 31.</td>
</tr>
<tr>
<td>--monthly-start-week=WEEK</td>
<td>Sets the monthly scheduled scan start week. The range of the week is first, second, third, fourth, and last.</td>
</tr>
<tr>
<td>--monthly-start-week-day=WEEKDAY</td>
<td>Sets the monthly scheduled scan start week day of the week. The range of the weekday is monday, tuesday, and so on.</td>
</tr>
</tbody>
</table>

**User example:**

To specify a monthly scheduled scan of a range of IP addresses with specified start time and day of the week and ends on completion of the scan:

```
tw_scan_control --monthly --monthly-start-week-day=monday --monthly-start-week=first --start-time=21:30 --no-end-time --add 192.168.0.1-10
```

**Backward compatible options**

Use the following backward compatible options with the `tw_scan_control` command line utility to add recurrent scans:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--recur-daily</td>
<td>Adds a daily recurrent range. This option specifies a recurrent range scan that must be modified with <code>recurrence-duration=int</code> and/or <code>recurrence-start=int</code>.</td>
</tr>
<tr>
<td>--recurrence-duration=INT</td>
<td>Specifies the duration for the recurring scan to last (in hours).</td>
</tr>
<tr>
<td>--recurrence-start=INT</td>
<td>Sets the start time for recurrent ranges (in hours) after midnight.</td>
</tr>
</tbody>
</table>

**Listing scheduled discovery runs**

Use the following common options with the `tw_scan_control` command line utility to list scheduled discovery runs:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--list</td>
<td>Lists all scan ranges.</td>
</tr>
<tr>
<td>--list-full</td>
<td>Lists all scan ranges with all IP addresses.</td>
</tr>
</tbody>
</table>

Listing the scheduled discovery runs gives you information about them, such as the range ID corresponding to a run, whether the run has been enabled or disabled, the label and IP addresses or ranges associated with a run, and so on.
User example:
To list all scan ranges with all IP addresses:

```
tw_scan_control --list-full
```

Updating and deleting scheduled discovery runs

Use the following common options with the `tw_scan_control` command line utility to update or delete scheduled discovery runs:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--clear</td>
<td>Cancels any running scans and deletes all scheduled scans.</td>
</tr>
<tr>
<td>--remove</td>
<td>Removes chosen scan ranges.</td>
</tr>
<tr>
<td>--update=ID</td>
<td>Updates (edit) the specified scheduled discovery run. The discovery run is specified using its range ID which can be determined by running the <code>list</code> or <code>list-full</code> options.</td>
</tr>
</tbody>
</table>

User examples:
To remove a chosen scan range:

```
tw_scan_control --remove 6ee6e7320aa7c5716c140a8148a76f8b
```

To update a chosen scheduled scan:

- Let us assume that you have set the following daily scheduled scan for an IP range where the start time is 14:30 and the end time is 17:30:

```
tw_scan_control --daily --start-time=14:30 --end-time=17:30 --add 192.168.0.1-10
```

Following are examples of updating the above scheduled scan.

- To update the start time to 20:30 and the end time to 23:50, you will run the following command:

```
tw_scan_control --daily --start-time=20:30 --end-time=23:50 --update=6ee6e7321031ae6a6fb80a8148a76f8b 192.168.0.1-10
```
To update it from a daily to a weekly scheduled scan, which starts on Monday at 07:30 and ends on Tuesday at 11:50, you will run the following command:

```
tw_scan_control --weekly --weekly-start-week-days=monday --weekly-end-week-day=tuesday --start-time=07:30 --end-time=11:50 --update=6ee6e732105205be37020a8148a76f8b
```

### Overlapping of scheduled scans and excludes

In the case of permanent excludes, discovery of the excluded endpoints never starts. However, in the case of an overlap of scheduled scans and scheduled excludes, the following behavior is expected:

- If a scheduled exclude overlaps with a scheduled scan, discovery of the excluded endpoints will not start until the scheduled exclude is no longer in effect.
- If the scheduled exclude ends before the scheduled scan end time, discovery of the excluded endpoints can start.
- If the scheduled exclude ends after the scheduled scan end time, the excluded endpoints will wait until the next time the scheduled scan runs.
- If one or more scheduled excludes overlap completely with a scheduled scan, the excluded endpoints will behave like permanent excludes. This is to prevent it from waiting forever to discover those excluded endpoints that it will never be able to scan.
- If a scheduled exclude is active and a snapshot scan is running, any excluded endpoint will be skipped by the scan and will have an excluded end state.

⚠️ **Deprecated utility: tw_injectip**

The `tw_injectip` utility is now deprecated and might be removed in future releases. Its functionality is available in the `tw_scan_control` utility which has all of the same options.

### `tw_reasoningstatus`

The `tw_reasoningstatus` utility enables you to view the status of the Reasoning service. Typically this utility is used by Customer Support as a troubleshooting tool for investigating possible problems.

⚠️ **Automatic use of tw_reasoningstatus**

Reasoning runs the same status check automatically every 15 minutes and outputs the results in the `tw_svc_reasoning.log` file.
To use the utility, type the following command at the $TIDEWAY/bin/ directory:

```
tw_reasoningstatus [options]
```

where *options* are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--waiting, -w</td>
<td>Lists information for all endpoints which are on hold waiting for information from the discovery of a different endpoint.</td>
</tr>
<tr>
<td>--waiting-full</td>
<td>Expands the information provided by --waiting to include information about all endpoints being held waiting for discovery. This option is ignored if --waiting is not specified.</td>
</tr>
<tr>
<td>-u, --username=NAME</td>
<td>Specifies the name of the BMC Atrium Discovery user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, you can view the status of the reasoning status for user *joe*.

**To view the status of the Reasoning service**

Type the following command:

```
$TIDEWAY/bin/tw_reasoningstatus --username joe
```

If you do not provide a password, you are prompted for one. After providing a password, a status is displayed that includes information about engine status, pool state, queue length, and so forth. The output is saved in the *tw_svc_reasoning.log* file.

**tw_model_init**

Deprecated. The recommended command is *tw_model_wipe*.

**tw_passwd**

The *tw_passwd* utility enables you to change the password of a specified user interface user. To use the utility, type the following command at the $TIDEWAY/bin/ directory:

```
tw_passwd [options] username
```
where:

- `username` is the name of the new user to change.
- `options` are the common, inherited commands described in Using command line utilities.

User example

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

Changing a user's password

```bash
$TIDEWAY/bin/tw_passwd joe
joeuser
```

⚠️ Changing passwords for command line users

The `tw_passwd` utility is for changing UI users' passwords. To change the passwords for command line users, as the root user, use the Linux command `passwd`. This is described in Changing the root and user passwords.

tw_upduser

The `tw_upduser` utility enables you to update properties of a specific user, such as the current state, permissions, and passwords. To use the utility, type the following command at the `$TIDEWAY` directory:

```bash
tw_upduser [options] username
```

where:

- `username` is the name of the user to update
- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--active</code></td>
<td>Sets the user to an ACTIVE state</td>
</tr>
<tr>
<td><code>--blocked</code></td>
<td>Sets the user to a BLOCKED state</td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>--disabled</code></td>
<td>Sets the user to a DISABLED state</td>
</tr>
<tr>
<td><code>-f, --fullname=ARG</code></td>
<td>Sets a full name of the user</td>
</tr>
<tr>
<td><code>-G, --groups=GROUP</code></td>
<td>Sets the user's group membership</td>
</tr>
<tr>
<td><code>--locked</code></td>
<td>Sets the user to a LOCKED state</td>
</tr>
<tr>
<td><code>--passwd-must-change</code></td>
<td>Specifies that the password must be changed</td>
</tr>
<tr>
<td><code>--passwd-ok</code></td>
<td>Specifies that the password is okay as currently stated</td>
</tr>
<tr>
<td><code>--passwd-should-change</code></td>
<td>Specifies that the password should be changed</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specifies the name of the BMC Atrium Discovery user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

### User example

In the following example, the `tw_upduser` utility sets the user `Joe` to a locked state to protect from unauthorized changes to that account.

#### Locking a user for security purposes

To lock a user, type the following command:

```
$TIDEWAY/bin/tw_upduser --locked Joe
```

The following confirmation is displayed:

```
Set User State USER_STATE_LOCKED
```

To unlock the user, run the utility again with the `--active` option.

### `tw_check_reports_model`

The `tw_check_reports_model` utility enables you to check custom reports for the following:

- Attributes marked as deprecated in the taxonomy
- Key expressions and traversals between Subnet nodes and Network Interface nodes. Subnet nodes are now related to IP Address nodes so these key expressions and traversals no longer exist.
- Usage of `ip_addr`, `netmask`, and `broadcast` attributes on Network Interface nodes. These attributes have been moved to IP Address nodes.
The utility is primarily intended for the upgrade to BMC Atrium Discovery 9.0, but it can also be used outside the upgrade. The username and password to supply for this utility are those of a UI user, such as the system user.

To use the utility, type the following command:

```
tw_check_reports_model [options]
```

where `options` are any of the common arguments described in Using command line utilities.

**User example**

The following example checks custom reports as the system user. No password is entered so the utility prompts for a password which is not echoed when entered.

**Checking for references to deprecated attributes**

```
[tideway@fredapp bin]$ tw_check_reports_model --username=system
Password:
Baseline.SLAM.Host-Solaris
    host_type has been deprecated on Host
Baseline.LogicalSystem.NetworkInterface
    ip_addr has moved from NetworkInterface to IPAddress
    netmask has moved from NetworkInterface to IPAddress
    broadcast has moved from NetworkInterface to IPAddress
Baseline.SLAM.Host-Linux
    host_type has been deprecated on Host
Baseline.LogicalSystem.SwitchPort
    Unknown node kind 'PortInterface'
Baseline.SLAM.Host-Windows
    host_type has been deprecated on Host
Baseline.SLAM.Host-Other
    host_type has been deprecated on Host
Baseline.SLAM.Host-AIX
    host_type has been deprecated on Host
[tideway@fredapp bin]$
```

This example has been formatted for easier reading.

**tw_adduser**

The `tw_adduser` utility enables you to add a new user to the system, and assign a user to a new group.
Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_adduser` command line utility (see Enabling other users). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_adduser [options] username
```

where:

- `username` is the name of the new user.
  If you do not specify a user name, BMC Atrium Discovery will use the default, system.

- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-f, --fullname=ARG</code></td>
<td>(Optional) Specifies the full name of the user</td>
</tr>
<tr>
<td><code>-g, --groups=ARG</code></td>
<td>(Optional) Specifies additional groups to add a user to (the default value is <code>public</code>). The value <code>public</code> is also the default when you select the Groups option from the Add User page on the Administration tab.</td>
</tr>
</tbody>
</table>

User example

In the following example, you can add a new administrative user to a multiple groups by using the utility, the `-g` option, and a comma-separated list.

**Adding a new user to multiple groups**

```
$TIDEWAY/bin/tw_adduser -g discovery,admin newuser
```

**tw_cron_update**

The `tw_cron_update` utility is used to manage tideway user cron.

**Warning**
Do not use `crontab -e` to edit the tideway user cron directly on the appliance.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_cron_update
```

**Cron overview**

BMC Atrium Discovery makes use of cron on the appliance to run various housekeeping tasks. This means that `crontab` must not be directly edited on the appliance, but must instead be managed using `tw_cron_update`. This way, BMC Atrium Discovery can manage cron entries across upgrades without affecting, or being affected by, any local customizations.

Cron entries are stored in `$TIDEWAY/etc/cron`. Each cron entry has its own file, using the file extension `.cron` to identify that file. A cron entry is created by adding a new file ending with `.cron` containing standard cron formatted commands. Similarly entries can be edited or removed by editing or removing `.cron` files.

`$TIDEWAY/bin/tw_cron_update` checks, every time it is run, that the file has not been edited since `tw_cron_update` was last run.

A note on the format of a cron entry is provided in the `tw_cron.header` file. For full details, see the Red Hat Enterprise Linux version 6 cron documentation.

⚠️ **Note**

If `crontab` has been directly edited, `tw_cron_update` can no longer manage cron. You will have to manually resolve any differences before continuing. A copy of the expected cron configuration is stored as `$TIDEWAY/etc/cron/tw_cron.previous` whenever `tw_cron_update` is run. This can be compared to the live configuration. If this file does not exist, then the utility has never been used and `crontab` should be used as the default.

**Managing cron entries**

Cron entry files in `$TIDEWAY/etc/cron` are applied to the live cron configuration by running `$TIDEWAY/bin/tw_cron_update` as the tideway user.
- Run `$TIDEWAY/bin/tw_cron_update` as the tideway user. Every time the script is run it confirms that the current live cron configuration matches what BMC Atrium Discovery expects it to be. If the configuration is as expected, there will be no output. If there is a discrepancy, an appropriate message will be displayed, and the discrepancy must be resolved before continuing.
- To add a new cron entry, add a new `.cron` file.
- To remove a cron entry, remove its `.cron` file, or modify the file extension to something other than `.cron`, (for example, `.disabled`.
- To modify a cron entry, edit its `.cron` file.
- Run `$TIDEWAY/bin/tw_cron_update` as the tideway user.

**tw_disco_export_platforms**

The `tw_disco_export_platforms` utility enables you to export the customizable platform scripts used by BMC Atrium Discovery so that they can be copied to another appliance, where they can subsequently be imported using `tw_disco_import_platforms`. After the scripts are imported, the corresponding platforms are displayed on the Administration => Discovery Platforms page.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_disco_export_platforms [options]
```

where options are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--default</code></td>
<td>Specifies to only export default platform scripts (not current scripts)</td>
</tr>
<tr>
<td><code>-o, --output=FILE</code></td>
<td>Specifies the name of the output file. The default name is <code>platforms.xml</code>.</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specifies the name of the BMC Atrium user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

A password is required to use the commands.

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Transferring custom commands to another appliance**

The following example illustrates how to transfer a command to an appliance (if you had previously customized the command):
$TIDEWAY/bin/tw_disco_export_platforms --password 'secret'
--output my_scripts_backup.xml

tw_restore

The `tw_restore` utility enables you to restore a BMC Atrium Discovery backup from a local directory on the appliance, an SSH enabled remote system, or a Windows share. The utility can be used on an appliance where the UI cannot be started. To use the utility, type the following command:

tw_restore [options]

where:

- **options** are any of the options described in the following table and the common command line options described in Using command line utilities.

⚠️ Common password options are described in this table

Common password options are described in this table to make it clear whether the password referred to is for the local or remote system.

⚠️ Restoring backed up system data when a baseline run is in progress results in critical baseline alerts

Baseline is run periodically by the BMC Atrium Discovery system. Should baseline be running when a restore is started critical baseline alerts may be reported. The restore is unaffected by these alerts. The automatic baseline check schedule can be modified using the `tw_cron_update`. In order to minimize the chance of alerts no new periodic baseline checks are performed during the restore.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--backup-dir=DIR</td>
<td>Specifies a remote backup directory relative to the user's home directory on SSH servers, and the top-level share directory in Windows shares. This option is required for SSH and Windows shares, unless the alternative <code>--unc-path</code> is specified for Windows shares.</td>
</tr>
<tr>
<td>--backup-local</td>
<td>Restore from a local backup in the $TIDEWAY/var/backup directory.</td>
</tr>
<tr>
<td>--backup-smb</td>
<td>Restore from a backup on a Windows share.</td>
</tr>
</tbody>
</table>
## Command Line Option

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--backup-ssh</td>
<td>Restore from a backup on a remote SSH server.</td>
</tr>
<tr>
<td>--email=EMAIL</td>
<td>Specify an email address to receive notification when the restore completes. This depends on email being correctly configured on the appliance.</td>
</tr>
<tr>
<td>--fix-interrupted</td>
<td>This option can be used where BMC Atrium Discovery has been interrupted by an unscheduled reboot or power failure. When you run <code>tw_restore</code> you are prompted to use this option if it is required.</td>
</tr>
<tr>
<td>--force</td>
<td>Force backup restore when services are down. This can omit some auditing of the action.</td>
</tr>
<tr>
<td>-h, --help</td>
<td>Display help on standard options and exit.</td>
</tr>
<tr>
<td>-H, --host=HOST</td>
<td>The hostname or IP address of the remote host.</td>
</tr>
<tr>
<td>-u, --username=NAME</td>
<td>The user to run the utility as. This has to be a valid BMC Atrium Discovery UI user such as the system user, not a username used to access the command line via SSH.</td>
</tr>
<tr>
<td>-p, --password=PASSWD</td>
<td>Specifies the password of the BMC Atrium Discovery user. If no password is specified, you are prompted for one. This is only relevant for utilities that have the --username option. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td>--passwordfile=FILE</td>
<td>Specifies a file from which the password is to be read. This is only relevant for utilities that have the --username option. This file is not encrypted, though you can set the file permissions to owner-only (<code>chmod 600 passwordfile.txt</code>) to restrict access to the file. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td>-P, --port=PORT</td>
<td>The port on the remote Remote SSH server.</td>
</tr>
<tr>
<td>--preserve-identity</td>
<td>Preserve the identity and the HTTPS configuration of the appliance rather than take on the identity from the restored backup.</td>
</tr>
<tr>
<td>--remote-password=PASSWORD</td>
<td>The password for the remote user on the remote system. For SSH servers or Windows shares.</td>
</tr>
<tr>
<td>--remote-user=USER</td>
<td>The username for the remote user on the remote system. For SSH servers or Windows shares.</td>
</tr>
<tr>
<td>-S, --share=SHARE</td>
<td>The share name of the remote Windows share. For example <code>sharename</code>. This option is required for Windows shares, unless the alternative --unc-path is specified.</td>
</tr>
<tr>
<td>--stop-services</td>
<td>Stop the services to perform the restore. Required when restoring from a backup.</td>
</tr>
<tr>
<td>--unc-path=PATH</td>
<td>The UNC path for the Windows share. For example <code>\sharename\foldername</code>. You can specify this as an alternative to --share and --backup-dir.</td>
</tr>
<tr>
<td>--verify-only</td>
<td>Verify the backup without performing a restore operation.</td>
</tr>
</tbody>
</table>

### User examples

In the following examples, the local username is system and the password is not specified on the command line. The remote SSH username is tideway and the password is password. The remote Windows share is called `remshare`, the corresponding username is `remdom/tideway` and the password is `password`.

The utility prompts for the local password after you enter the command. Type the commands on a single line; line breaks are provided in the examples to make them easier to read.
Verifying a local backup

```
$TIDEWAY/bin/tw_restore --verify-only --backup-local --username system
```

Verifying a remote backup on an SSH server

```
$TIDEWAY/bin/tw_restore --verify-only --backup-ssh --backup-dir=/usr/tideway/var/backup
   --remote-user=tideway --remote-password=password --host=de-32.bmc.com
   --username system
```

Verifying a remote backup on a Windows share

```
$TIDEWAY/bin/tw_restore --verify-only --backup-smb --share=remshare
   --backup-dir=2012-10-08_115740_addm_backup --remote-user=remdom/tideway
   --remote-password=password --host=192.168.1.101 --username system
```

Performing a restore from a local backup

```
$TIDEWAY/bin/tw_restore --stop-services --backup-local --username system
```

Performing a restore from an SSH server

```
$TIDEWAY/bin/tw_restore --stop-services --backup-ssh --backup-dir=/usr/tideway/var/backup
   --remote-user=tideway --remote-password=password --host=de-32.bmc.com --username system
```

Performing a restore from a Windows share

```
$TIDEWAY/bin/tw_restore --stop-services --backup-smb --share=remshare
   --backup-dir=2012-10-08_115740_addm_backup --remote-user=remdom/tideway
   --remote-password=password --host=192.168.1.101 --username system
```
tw_backup

The `tw_backup` utility enables you to create a backup of a BMC Atrium Discovery system on a local directory on the appliance, an SSH enabled remote system, or a Windows share. To use the utility, type the following command:

```
tw_backup [options]
```

where:

- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

⚠️ **Common password options are described in this table**

Common password options are described in this table to make it clear whether the password referred to is for the local or remote system.

‼️ **Backing up the system when a baseline run is in progress results in critical baseline alerts**

Baseline is run periodically by the BMC Atrium Discovery system. Should baseline be running when a backup is started critical baseline alerts may be reported. The backup is unaffected by these alerts. The automatic baseline check schedule can be modified using the `tw_cron_update`. In order to minimize the chance of alerts no new periodic baseline checks are performed during the backup.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--backup-dir=DIR</td>
<td>Specifies a remote backup directory relative to the user's home directory on SSH servers, and the top-level share directory in Windows shares. This option is required for SSH and Windows shares, unless the alternative --unc-path is specified for Windows shares.</td>
</tr>
<tr>
<td>--backup-local</td>
<td>Create a local backup in the $TIDEWAY/var/backup directory. Only one local backup can be stored.</td>
</tr>
<tr>
<td>--backup-smb</td>
<td>Create a backup on a Windows share.</td>
</tr>
<tr>
<td>--backup-ssh</td>
<td>Create a backup on a remote SSH server.</td>
</tr>
<tr>
<td>--email=EMAIL</td>
<td>Specify an email address to receive notification when the backup completes. This depends on email being correctly configured on the appliance.</td>
</tr>
</tbody>
</table>
### Command Line Option Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--exclude-sensitive</td>
<td>Exclude sensitive data from the backup. Appliance UI users are always backed up and restored, regardless of this setting.</td>
</tr>
<tr>
<td>--fix-interrupted</td>
<td>Unlock the system after a previous backup has failed to complete, starting services if possible.</td>
</tr>
<tr>
<td>--force</td>
<td>Force the creation of a backup when the services are down. This can omit some auditing of the action.</td>
</tr>
<tr>
<td>-h, --help</td>
<td>Display help on standard options and exit.</td>
</tr>
<tr>
<td>-H, --host=HOST</td>
<td>The hostname or IP address of the remote host.</td>
</tr>
<tr>
<td>--no-verification</td>
<td>Do not verify (md5) the backup after it has been created.</td>
</tr>
<tr>
<td>--notes=NOTES</td>
<td>Enter some free text notes about the backup.</td>
</tr>
<tr>
<td>--overwrite</td>
<td>Overwrite the existing local backup.</td>
</tr>
<tr>
<td>-u, --username=NAME</td>
<td>The user to run the utility as. This has to be a valid BMC Atrium Discovery UI user such as the system user, not a username used to access the command line via SSH.</td>
</tr>
<tr>
<td>-p, --password=PASSWD</td>
<td>Specifies the password of the BMC Atrium Discovery user. If no password is specified, you are prompted for one. This is only relevant for utilities that have the --username option. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td>--passwordfile=FILE</td>
<td>Specifies a file from which the password is to be read. This is only relevant for utilities that have the --username option. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td>-P, --port=PORT</td>
<td>The port on the remote Remote SSH server. Only required if the default SSH port (22) is not used.</td>
</tr>
<tr>
<td>--remote-password=PW</td>
<td>The password for the remote user on the remote system. For SSH servers or Windows shares.</td>
</tr>
<tr>
<td>--remote-user=USER</td>
<td>The username for the remote user on the remote system. For SSH servers or Windows shares. To specify the domain for Windows shares, use the following syntax: --remote-user=user@domain</td>
</tr>
<tr>
<td>-S, --share=SHARE</td>
<td>The share name of the remote Windows share. For example \sharename. This option is required for Windows shares, unless the alternative --unc-path is specified.</td>
</tr>
<tr>
<td>--stop-services</td>
<td>Stop services to create the backup. Required when creating a backup.</td>
</tr>
<tr>
<td>--unc-path=PATH</td>
<td>The UNC path for the Windows share. For example \sharename\foldername. You can specify this as an alternative to --share and --backup-dir.</td>
</tr>
<tr>
<td>-v, --version</td>
<td>Display version information and exit.</td>
</tr>
</tbody>
</table>

### User examples

In the following examples, the local username is system and the password is not specified on the command line. The remote SSH username is tideway and the password is \password. The remote Windows share is called remshare, the corresponding username is remdom/tideway and the password is \password.

The utility prompts for the local password after you enter the command. Type the commands on a single line; line breaks are provided in the examples to make them easier to read.
Create local backup

```
$TIDEWAY/bin/tw_backup --stop-services --backup-local --username=system
```

Create backup on a ssh enabled remote system

```
$TIDEWAY/bin/tw_backup --stop-services --backup-ssh --backup-dir=/addm_backups --remote-user=tideway --remote-password=password --host=de-32.bmc.com --username=system
```

tw_listusers

The `tw_listusers` utility enables you to display BMC Atrium Discovery user information, and lets you filter out specific names you do not want to display. To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_listusers [options] --filter=ARG
```

where:

- `--filter=ARG` specifies a filter (regular expression) for listing users. Only users whose username match this regular expression are listed.
- `options` are the common, inherited commands described in Using command line utilities.

User example

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

Filtering the list of users to just those whose username contains the string "joe"

```
$TIDEWAY/bin/tw_listusers --filter=joe
```
**tw_imp_csv**

The `tw_imp_csv` utility enables you to search the datastore for nodes of a specified kind that have keys matching rows in the supplied csv data. Where the keys match, the node is updated, or deleted and recreated (depending on the options selected).

---

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_imp_csv` command line utility (see Importing CSV Data). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

---

**Incorrect usage may result in data loss**

Before using the `tw_imp_csv` tool you should fully understand the system taxonomy and the changes that you are going to make to your data. Using the `tw_imp_csv` tool incorrectly can cause irreparable damage to your data. The data you submit using this tool is applied directly to the production data without any validation.

Always back up your datastore before using this tool.

---

**Do not import the following node kinds**

- You *must never* import DDD nodes.
- You should avoid importing Host nodes and other system maintained nodes. If in doubt, contact Customer Support.

---

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_imp_csv [options] files
```

where:

- `files` is a list of csv files to import.
• **options** are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--create</td>
<td>Specifies that only new nodes will be created; existing nodes are not updated</td>
</tr>
<tr>
<td>--datastore=name</td>
<td>(Do Not Use) Specifies the name of the datastore service. The default is tideway.</td>
</tr>
<tr>
<td>--delete</td>
<td>Specifies that matching nodes will be deleted; no other nodes will be affected</td>
</tr>
<tr>
<td>--delimiter=ARG</td>
<td>Specifies a field delimiter character</td>
</tr>
<tr>
<td>--escape-char=ARG</td>
<td>Specifies an escape character</td>
</tr>
<tr>
<td>--force</td>
<td>Disables the validation checks that are performed against the taxonomy to check the specified node-kind and attribute/relationship keys. When you use this option, all keys, attributes and relationship links will be left as strings.</td>
</tr>
<tr>
<td>--keys=keys</td>
<td>Displays a comma-separated list of key columns to use for the data key</td>
</tr>
<tr>
<td>--kind=kind</td>
<td>Displays the kind of node to create, update, or clear</td>
</tr>
<tr>
<td>--line-terminator=ARG</td>
<td>Specifies a line termination string</td>
</tr>
<tr>
<td>--partition=name</td>
<td>Specifies the name of the partition to query, where name can be set to All for all partitions (the default).</td>
</tr>
<tr>
<td>--quote-char=ARG</td>
<td>Specifies a quote character</td>
</tr>
<tr>
<td>--search=name</td>
<td>(Do Not Use) Specifies the name of the search service</td>
</tr>
<tr>
<td>--taxonomy=name</td>
<td>(Do Not Use) Specifies the name of the taxonomy service</td>
</tr>
<tr>
<td>--update</td>
<td>Specifies that only existing nodes will be updated; new nodes will not be created</td>
</tr>
<tr>
<td>--upload=UPLOAD</td>
<td>Specifies to hide the name of the uploaded file</td>
</tr>
<tr>
<td>--username=name-u</td>
<td>Specifies the BMC Atrium Discovery user name to use to import data. If no name is specified, you are prompted for one.</td>
</tr>
<tr>
<td>--verbose</td>
<td>Specifies to display informational messages while processing. This is useful for diagnosing errors</td>
</tr>
</tbody>
</table>

**User examples**

In the following examples, enter the commands on a single line. Line breaks are provided to make the examples easier to read.
Freeing rack space for applications

To free rack space for other applications, some hosts have been moved from the 'Campus' data centre to the newly acquired 'Firehouse' data centre. Discovery and Reasoning have handled the IP address and subnet changes but the Host nodes are still linked to the wrong location. Here is the CSV file to process, called firehouse_move.csv:

<table>
<thead>
<tr>
<th>name</th>
<th>HostInLocation:HostLocation:LocationOfHost:Location.name</th>
</tr>
</thead>
<tbody>
<tr>
<td>egon</td>
<td>Firehouse</td>
</tr>
<tr>
<td>ray</td>
<td>Firehouse</td>
</tr>
<tr>
<td>peter</td>
<td>Firehouse</td>
</tr>
</tbody>
</table>

BMC Atrium Discovery processes the CSV file with the following utility at the command line:

```
$ tw_imp_csv --username=system --password=system --kind=Host firehouse_move.csv
```

The utility reads the file called firehouse_move.csv line by line. It uses the first line to name the columns. The first column is called 'name', which doesn't begin with a '#' character so it is treated as an attribute name. The second column does begin with a '#' character so it is treated as a specification for some relationships.

No explicit key has been specified so the first (and in this case, only) attribute name is taken to be the key.

Next, the utility reads the second line. The first field, egon, is in the name column which was selected as the key earlier. So the script uses the search service to find a node of kind Host (from the --kind command line option) that has a name attribute equal to egon. It finds exactly one node matching that search. If it had not found that node, it would have been created. If it had found multiple nodes, an error would have been reported and processing would continue with the next line, NO nodes would have been updated.

Having found the node, it updates it using the other fields on the row. Were there any other attribute columns in the file, the utility would have used these to update the node before looking at the relationships.

The file has only one relationship column. The name is #HostInLocation:HostLocation:LocationOfHost:Location.name.

The utility searches for a Location node with a name attribute equal Firehouse, this row's value for the column. Having found the Location node, the utility creates a HostLocation relationship to it with the Host node playing the HostInLocation role and the Location playing the LocationOfHost role.
Then the utility processes the second and third data rows, updating the ray and peter nodes with the new location.

**Installing a remote Windows proxy**

A new host has been installed in the Firehouse data centre. Due to pressure from the organisation's E-services Protection Adviser, there is now a firewall preventing discovery of hosts on that site. Until a remote Windows proxy can be installed, the Firehouse system administrators have been sending us spreadsheets with the changes.

The CSV file `new_host.csv` looks like the following:

```
name,fqdn,#HostInLocation:HostLocation:LocationOfHost:Location.name,#ITOwnedItem:ITOwner:ITOwner:Person.name,ip_addrs,#HostOnSubnet:HostSubnet:Subnet.name
winston,winston.example.com,Firehouse,"Melnitz,J","['10.3.4.1','192.168.101.45']","['10.0.
```

BMC Atrium Discovery processes the CSV file with the following utility at the command line:

```
$ tw_imp_csv --username=system --password=system --kind=Host --create new_host.csv
```

The syntax is nearly the same as in the previous example. The differences are that the filename has changed because we are processing a different file and we are using Create Only mode.

As in the previous example, the utility searches for Host nodes with a name attribute equal to winston. This time, because it is in create mode, the utility checks that there are no matching nodes. If there were, the utility would report an error and skip the row.

Next, the utility creates a new node. It populates the attributes of the node from the non-relationship fields in the data. The ip_addrs field is a list and the value starts with a '[' character so it is converted into a list. The new node has attributes:

- name = 'winston'
- fqdn = 'winston.example.com'
- ip_addrs = [ '10.3.4.1', '192.168.101.45' ]

Then the utility adds the relationships. The location relationship column is processed in the same way as in Example 1. The column called #ITOwnedItem:ITOwner:ITOwner:Person.name creates an ITOwnedItem:ITOwner:ITOwner relationship to a Person node with a name equal to Melnitz,J. The quotes around that field are needed because the field contains a comma.
After the quote, the #HostOnSubnet:HostSubnet:Subnet.name field begins with a '[' character. This field is converted into a list. Then, for each item in the list, a HostOnSubnet:HostSubnet:Subnet relationship is created to a Subnet node with a name attribute equal to that item. So the new host has one relationship to the 10.0.0.0/8 subnet and one to the 192.168.101.0/24 subnet.

tw_imp_ciscoworks

The tw_imp_ciscoworks utility enables you to import CiscoWorks network device data from the command line.

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the tw_imp_ciscoworks command line utility (see Importing Network Device Data). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

To use the utility, type the following command at the $TIDEWAY/bin/ directory:

```
tw_imp_ciscoworks [options] --username name --file filename
```

where:

- **--username name** is the BMC Atrium Discovery user name to use to import data.
- --file filename is the name of the CiscoWorks data file to import.
- **options** are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--delimiter 'delimiter'</td>
<td>Specifies the field delimiter for input files. The default is the tab character.</td>
</tr>
<tr>
<td>--destroy-missing</td>
<td>Removes network devices and associated ports that do not exist in the imported data</td>
</tr>
<tr>
<td>--has-header</td>
<td>Specifies that the CiscoWorks data file to import has a header row</td>
</tr>
<tr>
<td>--no-header</td>
<td>Specifies that the CiscoWorks data file to import has no header row</td>
</tr>
<tr>
<td>--xml</td>
<td>Specifies that the data file being imported is in XML format, not CSV</td>
</tr>
</tbody>
</table>
User examples

In the following examples, enter the commands on a single line. Line breaks are provided to make the examples easier to read.

Generating and importing CSV files

To generate a CSV file on the CiscoWorks server, type the following:

```
C:\Program Files\CSCOpx\campus\bin\ut -cli -query all -layout StandardTideway -export c:\data.csv -u user -p password
```

This command produces a Java stack trace. This is a known issue and can be ignored. The file that is produced can be imported by running the following command on the BMC Atrium Discovery appliance:

```
$TIDEWAY/bin/tw_imp_ciscoworks --delimiter=',' --username name --password password --file ~/tmp/data.csv
```

Generating and importing XML files

To generate an XML file on the CiscoWorks server, type the following:

```
C:\Program Files\CSCOpx\campus\bin\cmexport ut -u userid -p password -host -layout StandardTideway
```

The file is written into the following directory:

```
C:\PROGRAM FILES\CSCOpx\files\cmexport\ut
```

The file can be imported onto the BMC Atrium Discovery appliance with the following command:

```
$TIDEWAY/bin/tw_imp_ciscoworks --xml --username name --password password --file ~/tmp/2006516154526ut.xml
```
The `tw_disco_import_platforms` utility enables you to import the customizable platform scripts used by BMC Atrium Discovery after they have been copied to another appliance, where they have been exported using `tw_disco_export_platforms`. After the scripts are imported, the corresponding platforms are displayed on the Administration => Discovery Platforms page.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```bash
$TIDEWAY/bin/tw_disco_import_platforms [options] files
```

where:

- `files` is a list of files to be imported (except for when you use the `reset` option).
- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--reset</code></td>
<td>Resets all commands to the default value. This is equivalent to clicking Reset All on the Administration =&gt; Platforms page (except only commands that have been customized will display).</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specify the name of the BMC Atrium user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Importing previously exported discovery scripts**

The following command imports previously exported scripts to an appliance:

```bash
$TIDEWAY/bin/tw_disco_import_platforms --password 'secret'
my_scripts_backup.xml
```

The `tw_baseline` utility enables you to audit your appliance for existing state, inventory, and relationships between configuration items.

**Recommendation**
Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_baseline` command line utility (see Baseline Configuration). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

Options in the utility specify how the baseline properties can be performed or displayed, or determine what action to take. To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_baseline [options]
```

where `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--conditional</td>
<td>Specifies to check for missing data and baselines only that data</td>
</tr>
<tr>
<td>--config=FILE</td>
<td>Specifies to baseline a configuration file, where <code>FILE</code> is the name of the file</td>
</tr>
<tr>
<td>--csvfile=FILE</td>
<td>Specifies to baseline a CSV output file, where <code>FILE</code> is the name of the file</td>
</tr>
<tr>
<td>--interactive</td>
<td>Specifies to run a baseline check in interactive mode, whereby you are prompted to respond Yes (Y) or No (N) to run specific checks until there are no further checks available. For checks that you request to run, results are displayed after you have responded to all checks. Checks that you have declined to run are skipped, and no further actions are run.</td>
</tr>
<tr>
<td>--no-actions</td>
<td>Specifies to run all baseline checks and perform no subsequent action after the checks are complete</td>
</tr>
<tr>
<td>--no-checks</td>
<td>Specifies to run the baseline and performs no subsequent checks</td>
</tr>
<tr>
<td>--no-display</td>
<td>Specifies to run the baseline without displaying results</td>
</tr>
<tr>
<td>--no-highlight</td>
<td>Specifies to run the baseline with no terminal highlighting</td>
</tr>
<tr>
<td>--rebaseline</td>
<td>Specifies to rebaseline the check data</td>
</tr>
<tr>
<td>--statusfile=FILE</td>
<td>Specifies to rebaseline a status output file, where <code>FILE</code> is the name of the file</td>
</tr>
<tr>
<td>--verbose</td>
<td>Specifies to run the baseline and display information messages for checked data</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.
Running a manual tripwire check

By default, Tripwire is run hourly and the output is written to the `tw_tripwire.txt` file. If a deviation from the baseline has been detected, the `tw_tripwire.txt` file is updated with the details. The monitor which sets the appliance status in the user interface checks the `tw_tripwire.txt` file hourly and sets certain restrictions if configured.

If you have rebaselined the Tripwire database, you should run the following commands (as the tideway user) to ensure that the correct status is shown in the user interface.

```bash
sudo /usr/tideway/tripwire/sbin/tripwire --check > /usr/tideway/var/tw_tripwire.txt
/usr/tideway/bin/tw_baseline
```

The appliance status is updated.

For more information about using Tripwire and baseline configuration, see [Baseline configuration](#).

**tw_tripwire_rebaseline**

The `tw_tripwire_rebaseline` utility enables you to rebaseline a Tripwire database. Tripwire is a third-party software tool that monitors a specific set of configuration, system, and source files on an appliance. When you use the utility to rebaseline the Tripwire database, you accept that all files that are being monitored are correct. To use the utility, type the following command at the directory:

```
tw_tripwire_rebaseline
```

You might use the script to initialize the Tripwire database, commission and configure Tripwire, or to run a reconfiguration procedure after the rebaseling process returns errors. For more information about rebaselining Tripwire and baseline configuration, see [Baseline configuration](#).

**User example**

In the following example, a user updates the Tripwire database after an error. This procedure should be performed as the tideway user.

**Updating the tripwire database after an error**

1. Check the items that are reported in the violation report and ensure that the reported changes are what you expected.
2. Run the following command:

```
/usr/tideway/bin/tw_tripwire_rebaseline
```
**tw_disco_control**

The `tw_disco_control` utility enables you to manually control specific functions of the BMC Atrium Discovery process. To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_disco_control [options] [opt=val ...]
```

where:

- `opt=val/...` is a list of options to be set.

- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

---

**Warning**

Use caution when using the `--emergency`, `--start`, and `--stop` options to start or stop BMC Atrium Discovery. Stopping discovery does not stop the Reasoning process. You must restart the discovery process before you can restart any scheduled scans.

---

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--devices</code></td>
<td>Specifies to display current devices</td>
</tr>
<tr>
<td><code>--emergency</code></td>
<td>Specifies to stop BMC Atrium Discovery immediately. All scheduled discovery scans are stopped.</td>
</tr>
<tr>
<td><code>--passphrase=ARG</code></td>
<td>Specifies the passphrase of the vault</td>
</tr>
<tr>
<td><code>--playback</code></td>
<td>Specifies to set BMC Atrium Discovery into playback mode</td>
</tr>
<tr>
<td><code>--quiet</code></td>
<td>Specifies that the user does not receive any informational messages</td>
</tr>
<tr>
<td><code>--record</code></td>
<td>Specifies to set BMC Atrium Discovery into recording mode</td>
</tr>
<tr>
<td><code>--standard</code></td>
<td>Specifies to set BMC Atrium Discovery into standard operating mode. It cannot be set to playback or record mode.</td>
</tr>
<tr>
<td><code>--start</code></td>
<td>Specifies to start BMC Atrium Discovery. All scheduled discovery scans are started.</td>
</tr>
<tr>
<td><code>--stop</code></td>
<td>Specifies to stop BMC Atrium Discovery. All scheduled discovery scans are stopped.</td>
</tr>
<tr>
<td><code>--test-cancel=ARG</code></td>
<td>Specifies to cancel a specified test</td>
</tr>
<tr>
<td><code>--test-remove=ARG</code></td>
<td>Specifies to remove a specified test</td>
</tr>
<tr>
<td><code>--tests</code></td>
<td>Specifies to display current tests</td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specifies the name of the BMC Atrium Discovery user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, you can stop a discovery scan in progress.

**Stopping the discovery process**

```
$TIDEWAY/bin/tw_disco_control
--stop
```

Any regular scans and snapshot scans currently in progress will stop, and no subsequent scans can be started until you restart the discovery process.

**tw_excluderanges**

The `tw_excluderanges` utility enables you to do the following:

- add permanent and scheduled excludes of IP addresses or ranges
- disable and enable excludes
- remove and update excludes

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_excluderanges` command line utility (see Running Discovery). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

This page contains the following sections:

- **Using the tw_excluderanges utility** -- This section contains the general guidelines to use the `tw_excluderanges` utility.
- **Common options to manage immediate and scheduled excludes** -- This section contains information about the common options to manage immediate and scheduled excludes, such as adding a new exclude, adding an exclude description, specifying a file, specifying a label, and so on.
• **Options to manage scheduled excludes** -- This section contains information about the options to manage scheduled excludes, such as adding scheduled exclude, enabling and disabling scheduled excludes, listing excludes, updating and deleting scheduled excludes, and so on.

• **Overlapping of scheduled scans and excludes** -- This section contains information about the expected behavior in the event of overlapping of scheduled scans and excludes.

• **Importing IP ranges to use as exclude ranges** -- This section contains information about importing multiple IP ranges from a text file to use as exclude ranges.

**Using the tw_excluderanges utility**

To use the utility, type the following command:

```
tw_excluderanges [options] args
```

where:

- **args** is one of the following arguments:
  - with **--disable**, **--enable** or **--remove**, a list of range IDs
  - with **-f** a list of filenames containing IP ranges to exclude
  - otherwise a list of IP ranges to exclude

If you do not select an argument in the command, a list of the currently excluded ranges is displayed, which includes the exclude range ID and additional information about that range. You could redirect this output to a file and then clean it up in a text editor to serve as a file which could then be imported.

**Common options to manage immediate and scheduled excludes**

⚠️ The common command line options are described in Using command line utilities.

Following are the common options for immediate and scheduled excludes with the **tw_excluderanges command line utility**:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a, --add</td>
<td>Adds a new exclude range.</td>
</tr>
<tr>
<td>-d, --description=DESCRIPTION</td>
<td>Specifies a description for the exclude range.</td>
</tr>
<tr>
<td>-f, --file</td>
<td>Specifies a file or a list of files as arguments. They must be plain text files with a new line delimited list of IP addresses. This is useful for importing large numbers of exclude ranges.</td>
</tr>
<tr>
<td>--label=LABEL</td>
<td>Specifies the label for the exclude range.</td>
</tr>
</tbody>
</table>
Command Line Option | Description
---|---
--silent | Turns off informational messages.

User examples:
To permanently exclude an IP range from discovery:

```
tw_excluderanges --add 192.168.0.1-10
```

To specify an exclude IP range listed in a file:

```
tw_excluderanges --add --file /tmp/ExcludeFile.txt
```

Options to manage scheduled excludes
The options to manage scheduled excludes with the `tw_excluderanges` command line utility enable you to perform the following:

- Enable and disable scheduled excludes
- Add scheduled excludes
- List scheduled excludes
- Update and delete scheduled excludes

Enabling and disabling scheduled excludes
Use the following common options with the `tw_excluderanges` command line utility to enable or disable scheduled scans:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--enable</td>
<td>Enables the chosen exclude ranges.</td>
</tr>
<tr>
<td>--disable</td>
<td>Disables the chosen exclude ranges.</td>
</tr>
</tbody>
</table>

User examples:
To enable a chosen exclude range:

```
tw_excluderanges --enable 6ee6e7321061632854040a8a76f8b
```

To disable a chosen exclude range:

```
tw_excluderanges --disable 6ee6e7321061632854040a8a76f8b
```
Adding scheduled excludes

Use the following common options with the `tw_excluderanges` command line utility to add scheduled excludes and specify its details:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--daily</td>
<td>Adds a daily scheduled exclude range.</td>
</tr>
<tr>
<td>--duration=DD:HH:MM</td>
<td>Sets the duration of a scheduled exclude.</td>
</tr>
<tr>
<td>--start-time=HH:MM</td>
<td>Sets the start time of a scheduled exclude.</td>
</tr>
<tr>
<td>--end-time=HH:MM</td>
<td>Sets the end time of a scheduled exclude.</td>
</tr>
<tr>
<td>--weekly</td>
<td>Adds a weekly scheduled exclude.</td>
</tr>
<tr>
<td>--weekly-start-week-days=WEEKDAYS</td>
<td>Sets the weekly scheduled exclude start week day of the week. The range of the weekday is monday, tuesday, and so on.</td>
</tr>
<tr>
<td>--weekly-end-week-day=WEEKDAY</td>
<td>Sets the weekly scheduled exclude end week day of the week. The range of the weekday is monday, tuesday, and so on.</td>
</tr>
<tr>
<td>--monthly</td>
<td>Adds a monthly scheduled exclude.</td>
</tr>
<tr>
<td>--monthly-start-day=DAY</td>
<td>Sets the monthly scheduled exclude start day. The range of the day is from 1 to 31.</td>
</tr>
<tr>
<td>--monthly-end-day=DAY</td>
<td>Sets the monthly scheduled exclude end day. The range of the day is from 1 to 31.</td>
</tr>
<tr>
<td>--monthly-start-week=WEEK</td>
<td>Sets the monthly scheduled exclude start week. The range of the week is first, second, third, fourth, and last.</td>
</tr>
<tr>
<td>--monthly-start-week-day=WEEKDAY</td>
<td>Sets the monthly scheduled exclude start week day of the week. The range of the weekday is monday, tuesday, and so on.</td>
</tr>
</tbody>
</table>

User examples:

To add a daily exclude IP range from discovery:

```
tw_excluderanges --daily --start-time=9:30 --end-time=11:30 --add 142.158.0.1-22
```

To specify a description for the exclude range:

```
tw_excluderanges --daily --start-time=9:30 --end-time=11:30 --description=TEST --add 142.158.0.1-22
```

To specify a duration for the exclude range:

```
tw_excluderanges --daily --start-time=9:30 --end-time=11:30 --add 142.158.0.1-22
```
To add a monthly exclude IP range from discovery:

```
tw_excluderanges --label=TEST --monthly --monthly-start-week-day=monday --monthly-start-week=first --start-time=21:30 --duration 00:06:30 --add 162.153.0.3-18
```

To add a weekly exclude IP range from discovery:

```
tw_excluderanges --weekly --weekly-start-week-days=monday --weekly-end-week-day=tuesday --start-time=21:30 --end-time=20:30 --add 123.142.0.6-13
```

Listing scheduled excludes

Use the following common options with the tw_excluderanges command line utility to list scheduled excludes:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--list</td>
<td>Lists all exclude ranges.</td>
</tr>
<tr>
<td>--list-full</td>
<td>Lists all exclude ranges with all IP addresses.</td>
</tr>
</tbody>
</table>

Listing the scheduled excludes gives you information about them, such as the range ID corresponding to an exclude, whether the exclude has been enabled or disabled, the label and IP addresses or ranges associated with a exclude, and so on.

User examples:

To list all exclude ranges with all IP addresses:

```
tw_excluderanges --list-full
```

Updating and deleting scheduled excludes

Use the following common options with the tw_excluderanges command line utility to update or delete scheduled excludes:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--clear</td>
<td>Removes all exclude ranges.</td>
</tr>
<tr>
<td>--update=ID</td>
<td></td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>Updates (edit) the specified scheduled exclude. The exclude is specified using its range ID which can be determined by running the <code>list</code> or <code>list-full</code> options.</td>
</tr>
<tr>
<td><code>-r</code>, <code>--remove</code></td>
<td>Removes chosen exclude ranges.</td>
</tr>
<tr>
<td><code>-x</code>, <code>--replace</code></td>
<td>If addresses supplied, adds a new exclude range, then delete all the old exclude ranges.</td>
</tr>
</tbody>
</table>

**User examples:**

To remove all exclude ranges from discovery:

```
tw_excluderanges --clear
```

To remove a chosen exclude range from discovery:

```
tw_excluderanges --remove 6ee6e73210ac294696f60a8148a76f8b
```

To update a chosen scheduled exclude:

Let us assume that you have set the following daily scheduled exclude for an IP range, where the start time is 14:30 and the end time is 17:30:

```
tw_excluderanges --daily --start-time=14:30 --end-time=17:30 --add 182.158.2.5-15
```

- To update the start time to 20:30 and the end time to 23:50, you will run the following command:

```
tw_excluderanges --daily --start-time=20:30 --end-time=23:50 --update=6ee6e73210b111bf7bec0a8148a76f8b 182.158.2.5-15
```

- To update it from a daily to a weekly scheduled scan, which starts on Monday at 07:30 and ends on Tuesday at 11:50, you will run the following command:

```
tw_excluderanges --weekly --weekly-start-week-days=monday --weekly-end-week-day=tuesday --start-time=07:30 --end-time=11:50 --update=6ee6e73210b111bf7bec0a8148a76f8b 182.158.2.5-15
```
Overlapping of scheduled scans and excludes

In the case of permanent excludes, discovery of the excluded endpoints never starts. However, in the case of an overlap of scheduled scans and scheduled excludes, the following behavior is expected:

- If a scheduled exclude overlaps with a scheduled scan, discovery of the excluded endpoints will not start until the scheduled exclude is no longer in effect.
- If the scheduled exclude ends before the scheduled scan end time, discovery of the excluded endpoints can start.
- If the scheduled exclude ends after the scheduled scan end time, the excluded endpoints will wait until the next time the scheduled scan runs.
- If one or more scheduled excludes overlap completely with a scheduled scan, the excluded endpoints will behave like permanent excludes. This is to prevent it from waiting forever to discover those excluded endpoints that it will never be able to scan.
- If a scheduled exclude is active and a snapshot scan is running, any excluded endpoint will be skipped by the scan and will have an excluded end state.

Importing IP ranges to use as exclude ranges

You can import multiple IP addresses or IPv4 ranges if they are contained in text files, one IP address or range per line. Ranges can be specified as usual:

- IPv4 address: for example 192.168.1.100.
- IPv6 address: for example fe80::655d:69d7:4bfa:d768.
- IPv4 range: for example 192.168.1.100-105, 192.168.1.100/24, or 192.168.1.*.

An example file called excludes1.txt:

```
192.168.1.100
192.168.1.110-120
```

A second example file called excludes2.txt:

```
192.168.2.100-105
192.168.2.*
192.168.3.0/24
```

Import the exclude ranges from the two files using the following command:
[tideway@appliance01 ~]$ tw_excluderanges --username=system --add
   --name="Imported Ranges" --file excludes1.txt excludes2.txt
Password:
Feeding file excludes1.txt
Feeding file excludes2.txt
Add excluded range: 192.168.1.100,192.168.1.110-120,192.168.2.100-105,
[tideway@appliance01 ~]$ tw_terminate_winproxy

The **tw_terminate_winproxy** utility enables you to send a request to the Windows proxy to terminate operation. To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_terminate_winproxy [options]
```

where *options* are any of the options described in the following table and the common command line options described in **Using command line utilities**.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--proxy-name proxyname</td>
<td><em>(Required)</em> Specifies the name of the Active Directory, Workgroup or Credential Windows proxy that you are logging into</td>
</tr>
<tr>
<td>--username username</td>
<td><em>(Required)</em> Specifies the name of the BMC Atrium Discovery user. If no name is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**Important**

You must have the `discovery/kslave/write` permission to use the utility.

In the following example, you must have the `discovery/kslave/write` permission to use the utility.

**Stopping a running credential Windows proxy**

```
$TIDEWAY/bin/tw_terminate_winproxy --proxy-name my_proxy
```
When the utility successfully sends a terminate request to a Windows proxy, an audit event is logged. The audit event, named `windows_proxy_process_terminate`, contains the name of the Windows proxy that the terminate request is sent to.

**Important**

If a Windows proxy is not running as either the Local System account or as a member of the Administrators group, `tw_terminate_winproxy` will fail to stop the Windows proxy. The following error is logged in the Windows proxy log file:

```
ERROR: Failed to terminate slave service: [(5, 'OpenSCManager', 'Access is denied.' )]
```

**Workaround:** Allow the user that the Windows proxy is running to stop the Windows proxy service. This is documented on the [Microsoft Support Site](https://support.microsoft.com).

For more information about Windows proxy configuration, see [Additional Windows proxy configuration](#).

**tw_deluser**

The `tw_deluser` utility enables you to delete a BMC Atrium Discovery user. To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_deluser [options] username
```

where:

- `username` is the name of the user to delete

- `options` are any of the common arguments described in [Using command line utilities](#).

**User example**

In the following example, a user deletes another user named `Joe` with a specific logging level of `info`.

**Deleting a user with a specific logging level**

```
$TIDEWAY/bin/tw_deluser --loglevel=info Joe
```
**tw_convert_reports**

The `tw_convert_reports` utility enables you to manually customize reports and channels on the appliance. This utility can serve as a standalone tool to manually convert reports from the old format to the new format (introduced in Tideway Foundation 7.3) after you have upgraded BMC Atrium Discovery and started the system.

This report format was introduced in Tideway Foundation 7.3. Therefore, the utility is required only for report files that have been manually copied from an appliance version earlier than 7.3 to an appliance version later than 7.3. If you upgrade to a BMC Atrium Discovery version later than 7.3, the utility runs automatically.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
$TIDEWAY/bin/tw_convert_reports options reports_file
```

where:

- `reports_file` is the name of the xml reports file to be converted. If a file name is not specified, an error is displayed.

- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-d, --debug</code></td>
<td>Specifies to display debug messages</td>
</tr>
<tr>
<td><code>-r, --rename</code></td>
<td>Specifies to rename the input file <code>reports_file.old</code> and save the converted file as <code>reports_file</code>. If there are errors, they are written to the terminal and the file is not converted.</td>
</tr>
<tr>
<td><code>-v, --verbose</code></td>
<td>Specifies to display errors and warning messages. If no errors are reported, the reports and charts work the same way they did in the previous version. These messages are also written in the comments section of the converted file.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Converting a reports file**

To convert a consolidated report file named `30ListenerReports.xml`, type the following command:

```
$TIDEWAY/bin/tw_convert_reports -d -v 30ListenerReports.xml
```
$ cd /usr/tideway/data/custom/reports
$ ls
30ListenerReports.xml
$ TIDEWAY/bin/tw_convert_reports 30ListenerReports.xml

Generated converted_30ListenerReports.xml (0 errors, 2 warnings)
$ ls
30ListenerReports.xml
converted_30ListenerReports.xml
$

The file is converted and saved as a new file prepended with converted_. The original file is not changed.

**tw_tax_export**

The `tw_tax_export` utility enables you to export taxonomy files so that they can be stored in another location. This utility can be used as an effective troubleshooting tool to check if all your taxonomy overlays worked as you expected, especially if you left old files in place or ordered them incorrectly.

Taxonomy definitions are configured using .xml files that are stored in the following directories:

/usr/tideway/data/installed/taxonomy/
/usr/tideway/data/custom/taxonomy/

The directories are parsed in the order shown (installed files take precedence over custom files), and the files contained in these directories are parsed in alphabetical order (numbers before letters). This order is important, because any taxonomy definitions that are subsequently added override any previously loaded definitions. The standard base taxonomy file is contained in `/usr/tideway/data/installed/taxonomy/00taxonomy.xml`.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_tax_export [options]
```

where *options* are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-D, --datastore=NAME</td>
<td><em>(Do Not Use)</em> Specifies the name of the datastore service</td>
</tr>
<tr>
<td></td>
<td>Specifies the name of the datastore partition</td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>-P, --</td>
<td>partition=NAME</td>
</tr>
<tr>
<td>-r, --</td>
<td>relkinds=REGEXP</td>
</tr>
<tr>
<td>-l, --</td>
<td>rolekinds=REGEXP</td>
</tr>
<tr>
<td>--sort</td>
<td>Specifies to sort names</td>
</tr>
<tr>
<td>-t, --taxonomy=LOC</td>
<td>(Do Not Use) Specifies the name of the taxonomy service</td>
</tr>
<tr>
<td>-u, --username=NAME</td>
<td>Specifies the name of the BMC Atrium Discovery user. If no user is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Exporting a taxonomy file**

```bash
$TIDEWAY/bin/tw_tax_export
/usr/tideway/data/installed/taxonomy/00taxonomy.xml
```

**tw_sign_winproxy_config**

The `tw_sign_winproxy_config` utility enables you to add a checksum to a Windows proxy configuration file without uploading the file. When you run this utility, it modifies the specified file but also saves a copy named `<original filename>.orig` before signing the file.

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_sign_winproxy_config` command line utility (see The Windows proxy configuration file). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_sign_winproxy_config [options]
```
where *options* are any of the options described in the following table and the common command line options described in [Using command line utilities](#).

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--config-file=PATHNAME</code></td>
<td>Specifies the name of the configuration file to sign.</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Displays the name of the BMC Atrium Discovery user. If a name is not specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Adding a checksum to a Windows proxy configuration file**

In the Checksum section of the Windows proxy configuration file, the checksum ensures that the file has not been tampered with since it has been copied or uploaded from the appliance. Using the `tw_sign_winproxy_config` utility, you can add the checksum without uploading the file to the appliance. To add the checksum to the file, type the following command:

```
$TIDEWAY/bin/tw_sign_winproxy_config --config-file=C:\Program Files\BMC Software\ADDM Active Directory Proxy\etc\winproxy.conf
```

You can then copy the signed file to multiple appliances using `ftp` or similar.

**tw_query**

The `tw_query` utility enables you to extract data using a query. The information can be output in CSV or XML format using one of the available arguments.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_query [options] query
```

where:

- *query* is the data you want to extract from the Search service.
- *options* are any of the options described in the following table and the common command line options described in [Using command line utilities](#).
tw_config_dashboards

The `tw_config_dashboards` utility enables you to configure and customize dashboards in BMC Atrium Discovery.

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionality provided by the `tw_config_dashboards` command line utility (see [Using and Customizing Dashboards](#)). If you choose to run the utility, read the documentation in this section to learn its usage and to understand the risks and potential impact on your environment.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_config_dashboards [options] title
```

where:

- `title` is the title of the dashboard.
  If a title is not specified, an error is displayed.

- `options` are any of the options described in the following table and the common command line options described in [Using command line utilities](#).

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--lock</td>
<td>Specifies to lock the dashboard so that it is read-only (the default)</td>
</tr>
<tr>
<td>--ls</td>
<td></td>
</tr>
</tbody>
</table>
## Command Line Option

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--add</td>
<td>Specifies to display a list of the current dashboards and whether they are locked or unlocked. For this option only, a title does not need to be specified.</td>
</tr>
<tr>
<td>--unlock</td>
<td>Specifies to unlock the dashboard so that it can be edited</td>
</tr>
</tbody>
</table>

### User example

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

#### Changing a dashboard

By default, dashboards are configured to be read-only. If you want to customize a dashboard, you first need to make it accessible using the --unlock option:

```bash
$TIDEWAY/bin/tw_config_dashboards --unlock Default
```

### tw_sync_control

The `tw_sync_control` utility enables you to control CMDB sync activities from the command line. The utility is primarily intended to be used as a means of automating certain CMDB sync procedures rather than an all purpose controller for CMDB sync. To use the utility, type the following command:

```
tw_sync_control [options]
```

where:

- **options** are any of the options described in the following table and the common command line options described in Using command line utilities.

#### Common password options are described in this table

Common password options are described in this table to make it clear whether the password referred to is for the local or remote system.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--add</td>
<td></td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>tw_sync_control --add --name=Fred42 --endpoint=192.168.1.100 --target-user=gorwell --target-password=1984 --continuous --dataset=BMC.ADDM --datamodel=6</td>
<td></td>
</tr>
<tr>
<td><strong>--continuous</strong></td>
<td>Enables continuous synchronization for the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--datamodel=DATAMODEL</strong></td>
<td>Target datamodel number (2-6). See Data models.</td>
</tr>
<tr>
<td><strong>--dataset=DATASET</strong></td>
<td>Name of dataset (default = BMC.ADDM).</td>
</tr>
<tr>
<td><strong>--disable</strong></td>
<td>Disable the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--enable</strong></td>
<td>Enable the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--endpoint=ENDPOINT</strong></td>
<td>The IP address or hostname of the BMC Atrium CMDB server.</td>
</tr>
<tr>
<td><strong>--export-filter</strong></td>
<td>Export filter definition for the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--filter-def=FILE</strong></td>
<td>Specify a filter definition file to write (--export-filter) or read (--import-filter).</td>
</tr>
<tr>
<td><strong>--id=TARGET_ID</strong></td>
<td>Specify the identifier of the CMDB sync connection. Determine the identifier using the --list option.</td>
</tr>
<tr>
<td><strong>--import-filter</strong></td>
<td>Import a filter definition --filter-def for the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--list</strong></td>
<td>Display a list of configured CMDB sync connections, together with the configuration information supplied when the connection was added or last updated.</td>
</tr>
<tr>
<td><strong>--name=NAME</strong></td>
<td>Specify the name of a connection. Used when adding a new connection.</td>
</tr>
<tr>
<td><strong>--not-continuous</strong></td>
<td>Disable continuous sync on the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--no-export</strong></td>
<td>Specify that a new connection should not actually export to a CMDB, but just maintain the local shadow copy of the data within BMC Atrium Discovery.</td>
</tr>
<tr>
<td><strong>--password=PASSWD</strong></td>
<td>Specifies the password of the BMC Atrium Discovery UI user specified using the --username option. If no password is specified, you are prompted for one. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td><strong>--passwordfile=PWDFILE</strong></td>
<td>Specifies a file from which the password for the user specified using the --username option is to be read. This is only relevant for actions that require the --username option. This file is not encrypted, though you can set the file permissions to owner-only (chmod 600 passwordfile. txt) to restrict access to the file. For more information about password policies, see Managing Security Policies.</td>
</tr>
<tr>
<td><strong>--pause</strong></td>
<td>Pause synchronization for the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--port=PORT</strong></td>
<td>Port of target server (default = 0). Used with the --endpoint option.</td>
</tr>
<tr>
<td><strong>--remove</strong></td>
<td>Remove the specified (--id) connection.</td>
</tr>
<tr>
<td><strong>--reset-status</strong></td>
<td>Reset the counts of nodes inserted, updated, and deleted.</td>
</tr>
<tr>
<td><strong>--rpc-queue=PORT</strong></td>
<td>RPC queue port of CMDB (default = 0). Used when configuring a private queue.</td>
</tr>
<tr>
<td><strong>--status</strong></td>
<td>Show status of all connections, including recently synced devices. The creates, updates, deletes, and errors counts are cumulative.</td>
</tr>
<tr>
<td><strong>--sync=QUERY</strong></td>
<td>Sync root nodes selected by query.</td>
</tr>
</tbody>
</table>
Command Line Option | Description
--- | ---
--target-password=PASSWD | The password of the BMC Atrium CMDB target user specified using the --target-user option.
--target-user=USER | The username for the BMC Atrium CMDB target.
--test-cred | Test credential.
--unpause | Unpause synchronization for the specified (--id) connection.
--update | Update the configuration of the specified (--id) connection.
--username=NAME | The user to run the utility as. This has to be a valid BMC Atrium Discovery UI user such as the system user, not a username used to access the command line via ssh.

Resync commands | See To resync a connection for examples of using these commands.
--- | ---
--resync-abort | Initiate resync abort for the specified (--resync-id) on the specified (--id) connection. This can be done after the resync has been committed.
--resync-cancel | Cancel resync for the specified (--resync-id) on the specified (--id) connection. Cannot be used once a resync has been committed.
--resync-commit | Initiate resync commit for the specified (--resync-id) on the specified (--id) connection.
--resync-diff | Initiate resync diff on the specified (--id) connection.
--resync-id=RESYNC_ID | Specify the ID of a resync. The ID is obtained from --resync-diff.
--resync-status | Show connection resync status.

Data models

CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name &quot;ImpactOnly&quot; are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name &quot;IMPACT&quot; are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name &quot;IMPACT&quot; are created.</td>
</tr>
</tbody>
</table>

User examples

To modify an existing connection

You can modify the configuration of an existing CMDB connection using the --update option:
The most likely usage of the `tw_sync_control` utility is to automate regular synchronization of an existing CMDB connection using cron updated with `tw_cron_update`. The following command is modified to use a password file and a query for the same connection:

```
[tideway@appliance01 ~]$ tw_sync_control --update --id 2d104b55361df7974e4d1bf86e434d444253796e63436f6e666967 --username=system --name 8102_4
Password: 
Target successfully updated:

   Id: 2d104b55361df7974e4d1bf86e434d444253796e63436f6e666967
   Name: 8102_4
   Type: CMDB
   Endpoint: <IP address>
   User: Demo
   Dataset: John_4
   Datamodel: CMDB 7.6.03 and later (with impact attributes)
   Continuous: False
```

```
[tideway@appliance01 ~]$ tw_sync_control --id 2d104b55361df7974e4d1bf86e434d444253796e63436f6e666967 --username=system --passwordfile=/usr/tideway/etc/passwd.txt --sync="search Host where name has subword 'example'"
```

The automation requires the command to be entered into a `.cron` file in the `/usr/tideway/etc/cron` directory. In this example, `tw_sync_control.cron`.

```
[tideway@appliance01 ~]$ cd etc/cron
[tideway@appliance01 cron]$ ls
   tw_appliance_stats.cron  tw_compress_logs.cron  tw_cron.header
   tw_cron.previous  tw_log_performance.cron  tw_sync_control.cron
[tideway@appliance01 cron]$ more tw_sync_control.cron
#
# cron job to sync Windows hosts
# runs every day at 01:00
#
0 1 * * * /usr/tideway/bin/tw_sync_control
   --id 2d104b55361df7974e4d1bf86e434d444253796e63436f6e666967 --username=system
   --passwordfile=/usr/tideway/etc/passwd.txt --sync="search Host where name has subword 'exa"
```

```
[tideway@appliance01 cron]$ cd etc/cron
```
To determine connection information

To determine the IDs of current connections

You use the connection ID to specify the particular connection on which to perform sync operations. To determine this:

```
[tideway@appliance01 ~]$ tw_sync_control --list --username=system
Password:
Sync targets:

    Id: 2d104b55361df7974e4d1bf86e434d444253796e63436f6e666967
    Name: 8102_4
    Type: CMDB
    Endpoint: <IP address>
    User: Demo
    Dataset: John_4
    Datamodel: CMDB 7.6.03 and later (with impact attributes)
    Multitenancy: False
    Available : False
    Continuous: False
    Blackouts: 0 blackout windows

    Id: 1b384a55302402bf596e8d226e434d444253796e63436f6e666967
    Name: ValidIP_SyncIP
    Type: CMDB
    Endpoint: <IP address>
    User: Demo
    Dataset: mltest
    Datamodel: CMDB 7.6.03 and later (with impact attributes)
    Multitenancy: False
    Available : Built In-House, Calbro Services, Invention, Inc.
    Continuous: False
    Blackouts: 0 blackout windows
```

To determine the status of a particular connection

To obtain the detailed status on a connection you must use the connection ID reported when using the --list option, described above.

```
[tideway@appliance01 ~]$ tw_sync_control --status
--id=1b384a55302402bf596e8d226e434d444253796e63436f6e666967 --user=system
Password:
Sync target status at 2015-05-07 16:13:31

    Id: 1b384a55302402bf596e8d226e434d444253796e63436f6e666967
```
Basic operations on a connection

To pause or unpause a particular connection
To pause or unpause a connection you must use the connection ID reported when using the --list option, described above.

```
[tideway@appliance01 ~]$ tw_sync_control --pause
--id=1b384a55302402bf596e8d226e434d444253796e63436f6e666967 --username=system
Password:
Successfully paused sync target
[tideway@appliance01 ~]$ tw_sync_control --unpause
--id=1b384a55302402bf596e8d226e434d444253796e63436f6e666967 --username=system
Password:
Successfully unpaused sync target
```

To enable or disable a connection
To enable or disable a connection you must use the connection ID reported when using the --list option, described above.

```
[tideway@appliance01 ~]$ tw_sync_control --disable
--id=1b384a55302402bf596e8d226e434d444253796e63436f6e666967 --username=system
Password:
Successfully disabled sync target
[tideway@appliance01 ~]$ tw_sync_control --enable
--id=1b384a55302402bf596e8d226e434d444253796e63436f6e666967 --username=system
Password:
Successfully enabled sync target
```
To resync a connection

This is a multi-stage process:

1. Perform a resync diff on a specified connection so that BMC Atrium Discovery can determine any differences between its view of the world and the corresponding CMDB dataset. A resync ID is generated which is used for any further operations on that resync.
2. Commit the resync, that is, update the CMDB dataset with just the differences between it and the BMC Atrium Discovery view.
3. Perform a resync status to determine whether the resync has completed and whether there were any errors.

It is useful to perform a check on the resync status after each step. This is shown below.

```
[tideway@appliance01]$ tw_sync_control --resync-diff
--id=6ee56054621792fb04cf8f546e434d444253796e63436f6e666967
--username=system
Password:
Successfully initiated resync diff
resync-id = 35029632d0c0038cf28289485e524261
[tideway@appliance01]$ tw_sync_control --resync-status
--id=6ee56054621792fb04cf8f546e434d444253796e63436f6e666967
--username=system
Password:
Resync Status
  Id: 35029632d0c0038cf28289485e524261
  Status: WAITING USER COMMIT / CANCEL
  Tasks:
  Retrieve Dataset Data
    COMPLETED
    Retrieved 66 nodes, 96 relationships
  Update Sync Partition
    COMPLETED
    Processed 6 devices
  Evaluate Differences
    COMPLETED
    Evaluated 66 nodes, 96 relationships
  Total Counts:
    Create 0 Nodes
    Update 66 Nodes
    Delete 0 Nodes
    Create 0 Relationships
    Update 96 Relationships
    Delete 0 Relationships
  Kind Counts:
    Update 6 BMC_ComputerSystem Nodes
    Update 3 BMC_IPConnectivitySubnet Nodes
```
Update 11 BMC_IPEndpoint Nodes
Update 14 BMC_LANEndpoint Nodes
Update 20 BMC_NetworkPort Nodes
Update 4 BMC_OperatingSystem Nodes
Update 4 BMC_Processor Nodes
Update 2 BMC_Product Nodes
Update 2 BMC_SoftwareServer Nodes
Update 30 BMC_Dependency Relationships
Update 25 BMC_HostedAccessPoint Relationships
Update 30 BMC_HostedSystemComponents Relationships
Update 11 BMC_InIPSubnet Relationships

[tideway@appliance01]$ tw_sync_control --resync-commit
--resync-id=35029632d0c0038cf28289485e524261
--id=6ee56054621792f04cf8f546e434d444253796e63436f66967 --username=system
Password:
Successfully initiated resync commit

[tideway@appliance01]$ tw_sync_control --resync-status
--resync-id=35029632d0c0038cf28289485e524261
--id=6ee56054621792f04cf8f546e434d444253796e63436f66967 --username=system
Password:
Resync Status
   Id: 35029632d0c0038cf28289485e524261
   Status: COMPLETED
Tasks:
   Retrieve Dataset Data
       COMPLETED
       Retrieved 66 nodes, 96 relationships
   Update Sync Partition
       COMPLETED
       Processed 6 devices
   Evaluate Differences
       COMPLETED
       Evaluated 66 nodes, 96 relationships
   Total Counts:
       Create  0 Nodes
       Update 66 Nodes
       Delete  0 Nodes
       Create  0 Relationships
       Update 96 Relationships
       Delete  0 Relationships
   Kind Counts:
       Update  6 BMC_ComputerSystem Nodes
       Update  3 BMC_IPConnectivitySubnet Nodes
       Update 11 BMC_IPEndpoint Nodes
       Update 14 BMC_LANEndpoint Nodes
       Update 20 BMC_NetworkPort Nodes
       Update  4 BMC_OperatingSystem Nodes
       Update  4 BMC_Processor Nodes
Update 2 BMC_Product Nodes
Update 2 BMC_SoftwareServer Nodes
Update 30 BMC_Dependency Relationships
Update 25 BMC_HostedAccessPoint Relationships
Update 30 BMC_HostedSystemComponents Relationships
Update 11 BMC_InIPSubnet Relationships

Commit Changes
COMPLETED
Processed 66 nodes, 96 relationships

Total Counts:
Created 0 Nodes
Updated 66 Nodes
Deleted 0 Nodes
Created 0 Relationships
Updated 96 Relationships
Deleted 0 Relationships

[tideway@appliance01]$ tw_sync_control --resync-cancel
--resync-id=35029632d0c0038cf28289485e524261
--id=6ee56054621792fb04cf8f546e434d444253796e63436f666967 --username=system
Password:
Successfully cancelled resync

[tideway@appliance01]$

tw_tax_import

The tw_tax_import utility enables you to import custom taxonomy files into the current taxonomy.

 Recommendation

After you run a taxonomy import using this utility, you must restart the tideway service. Failure to do so will result in a malfunctioning user interface.

Taxonomy definitions are configured using .xml files that are stored in the following directories:
The directories are parsed in the order shown (installed files take precedence over custom files), and the files contained in these directories are parsed in alphabetical order (numbers before letters). This order is important, because any taxonomy definitions that are subsequently added override any previously loaded definitions. The standard base taxonomy file is contained in `/usr/tideway/data/installed/taxonomy/00taxonomy.xml`.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_tax_import [options] files
```

where:

- `files` is a list of files to be imported. This command is optional if you use the `--clear` option, but cannot be used with the `--handle-broken-extensions` option.

- `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--clear</code></td>
<td>Specifies to clear any existing taxonomy before importing</td>
</tr>
<tr>
<td><code>--handle-broken-extensions</code></td>
<td>Specifies to ignore any broken extensions. If a taxonomy extension is invalid, the importer ignores it and restarts, attempting to load all other extensions. It repeats this until it has loaded the base taxonomy and has loaded or attempted to load all extensions. In this way you always finish with a valid taxonomy. Where the taxonomy importer fails to load an extension it logs a message to stdout. This option cannot be used with the <code>files</code> command. It is primarily intended for upgrading, but it can be used from the command line.</td>
</tr>
<tr>
<td><code>-D, --datastore=NAME</code></td>
<td>(Do Not Use) Specifies the name of the datastore service</td>
</tr>
<tr>
<td><code>--merge</code></td>
<td>Specifies to merge import data with any existing taxonomy</td>
</tr>
<tr>
<td><code>-P, --partition=NAME</code></td>
<td>Specifies the name of the datastore partition</td>
</tr>
<tr>
<td><code>--replace</code></td>
<td>Specifies to replace data in any existing taxonomy with imported data</td>
</tr>
<tr>
<td><code>--strict</code></td>
<td>Specifies that there is no backwards compatibility for the previous format</td>
</tr>
<tr>
<td><code>-t, --taxonomy=ARG</code></td>
<td>(Do Not Use) Specifies the name of the taxonomy service</td>
</tr>
<tr>
<td><code>-u, --username=NAME</code></td>
<td>Specifies the name of the BMC Atrium Discovery user. If no user is specified, BMC Atrium Discovery uses the default, system.</td>
</tr>
<tr>
<td><code>--verbose</code></td>
<td>Specifies to display informational messages</td>
</tr>
<tr>
<td>Command Line Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>--verify</td>
<td>Specifies to verify XML data only</td>
</tr>
</tbody>
</table>

**User example**

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

**Importing customized taxonomy definitions**

The following example illustrates how to import a custom taxonomy file and merge the imported data into the existing taxonomy.

1. Type the following command:

```bash
$TIDEWAY/bin/tw_tax_import --merge /usr/tideway/data/custom/taxonomy/MYtaxonomy.xml
```

The standard base taxonomy file named `/usr/tideway/data/installed/taxonomy/00taxonomy.xml` is supplemented with the imported data, and you can view the updated `installed.xml` file on the **Administration > Taxonomy** page.

2. Restart the tideway service.

**`tw_ds_offline_compact`**

Over time, the database files within the datastore can become fragmented, meaning that the data within them is structured inefficiently, so the files take up an unnecessarily large amount of disk space, and data access speed can suffer. The `tw_ds_offline_compact` utility enables you to compact the datastore by writing new copies of the database files. As it creates new files, the data is packed more efficiently, helping to alleviate lost disc space caused by fragmentation.

The tool operates on an **offline** system, meaning that the tideway services must be stopped when you use this tool.

Compacting the datastore might take a long time. In a large datastore, this can be many hours. Once a compaction starts, you should not interrupt it. To prevent loss of terminal connection interrupting the compaction, you should run `tw_ds_offline_compact` inside the **screen** utility which is installed on the appliance. The **user example** below shows how to do this.

To use the utility on a cluster, ideally you should stop the tideway services on all machines in the cluster and run `tw_ds_offline_compact` on each machine and then restart the tideway services when all compactions have completed. This method has the advantage that all machines can be compacted at the same time and the elapsed time is minimized. Alternatively, if you are running a cluster of three or more with **fault tolerance** enabled, and cannot stop your cluster, you
could stop the tideway services on a single machine, run `tw_ds_offline_compact` on that machine, and then restart the tideway services. Once the machine is working again in the cluster, you can perform the compaction on the remaining machines in turn. This method has disadvantages, compactions run individually in turn, so the elapsed time spent in compaction is much greater. Additionally, once a machine has been compacted, it needs to catch up with the activity that it missed while offline. In a busy cluster, this could take considerable time, depending on the cluster size, datastore size, and performance.

To use the utility, enter the following command:

```
tw_ds_offline_compact [options]
```

where *options* are any of the options described in the following table and the common command line options described in Using command line utilities.

### Warning

Because the utility touches all of the data, you *must* create a backup of the datastore before running it.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--fix-interrupted</code></td>
<td>Fix the databases following an interrupted compaction. See recovering from an interrupted compaction.</td>
</tr>
<tr>
<td><code>--no-questions</code></td>
<td>Proceed immediately with no questions.</td>
</tr>
<tr>
<td><code>--help</code></td>
<td>Displays help messages and exits.</td>
</tr>
</tbody>
</table>

#### User example

The following example shows backing up a datastore and then compacting it using the `tw_ds_offline_compact` utility.

**Backup the datastore**

Back up the datastore using the appliance backup tool.

**Compact the datastore**

1. Login to the appliance command line interface as the tideway user.
2. Run the screen utility. Enter:

```
[tideway@appliance01 ~]$ screen
```
3. Stop the tideway services.

   [tideway@appliance01 ~]$ sudo /sbin/service tideway stop

4. Run the `tw_ds_offline_compact` utility. Enter:

   [tideway@appliance01 ~]$ tw_ds_offline_compact

   The utility checks that the tideway services have been stopped and that there is sufficient disk space to continue. It then reports the data directory, largest database file, and the free space available. Before proceeding you must confirm that you have made a backup of the datastore.

5. Enter `yes` to confirm that you have made a backup of the datastore.

6. Enter `yes` to start the compaction.

7. Start the tideway services. Enter the following command:

   [tideway@appliance01 ~]$ sudo /sbin/service tideway start

---

**Recovering from a lost connection using screen**

If you lose the connection to the appliance and you have used screen, you can reconnect to the appliance and recover the virtual terminal running the compaction. To do this:

1. Reconnect to the appliance and login as the tideway user.
2. List the current screen sessions. Enter:

   [tideway@appliance01 ~]$ screen -ls
   There is a screen on:
   23274.pts-0.appliance01        (Detached)
   1 Socket in /var/run/screen/S-tideway.

   If there is only one screen listed, you can re-attach to it with a simple command:
If there is more than one, copy the screen identifier:

```
[tideway@appliance01 ~]$ screen -r
```

The virtual terminal is recovered and you can see how the compaction is progressing until completion.

3. Start the tideway services. Enter the following command:

```
sudo /sbin/service tideway start
```

Recovering from an interrupted compaction

If the compaction has been interrupted in some way, then the database files are left in a partial state and the datastore cannot run. You can recover from this situation in the following ways:

- Perform the compaction using the `tw_ds_offline_compact` utility again and allow it to complete without interruption. See the procedure above.
- Run the `tw_ds_offline_compact` utility again with the `--fix-interrupted` option. This fixes the datastore but does not perform any more compaction.

**tw_tax_deprecated**

The `tw_tax_deprecated` utility enables you to check patterns for references to deprecated attributes. The utility is primarily intended for the upgrade to BMC Atrium Discovery 9.0, but it can also be used outside the upgrade. The username and password to supply for this utility are those of a UI user, such as the system user.

To use the utility, type the following command:

```
tw_tax_deprecated [options]
```

where *options* are any of the common arguments described in Using command line utilities, or the command described in the following table:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--show-tku</code></td>
<td>Specifies to check for deprecated attributes in all installed TKU pattern modules.</td>
</tr>
</tbody>
</table>
User example

The following example checks patterns for references to deprecated attributes as the system user and includes all installed TKU pattern modules. No password is entered so the utility prompts for a password which is not echoed when entered. In this example, deprecated attributes are found.

Checking for references to deprecated attributes

```
[tideway@fredapp bin]$ tw_tax_deprecated --show-tku --username=system
Password:
Searching for deprecated Host node attribute 'host_type' ...

Found in pattern module:
HostLocations (Package host_locations)

Searching for deprecated NetworkDevice node attribute 'device_type' ...

Searching for deprecated MFPart node attribute 'part_type' ...

[tideway@fredapp bin]$```

This example has been formatted for easier reading.

tw_top

The `tw_top` utility provides a filter on the `top` utility which enables you to view just BMC Atrium Discovery processes. The `top` utility displays and updates a list of processes, ordered by default by CPU usage. You can use other sorting and display criteria. For more information see the `top` documentation, man page, or when the utility is running press ? to display help.

Tip

The process and service command lines are long so it works best with a wider terminal window.

Options in the utility specify which processes and services to display. To use the utility, type the following command:

```
tw_top [options]
```

where `options` are any of the options described in the following table and the common command line options described in Using command line utilities.
### Command Line Option

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-c, --cluster_manager</td>
<td>Show cluster manager processes.</td>
</tr>
<tr>
<td>-d, --discovery</td>
<td>Show the discovery service.</td>
</tr>
<tr>
<td>-m, --model</td>
<td>Show the model service.</td>
</tr>
<tr>
<td>-p, --providers</td>
<td>Show all providers.</td>
</tr>
<tr>
<td>-r, --reasoning</td>
<td>Show the reasoning service including ECA engines.</td>
</tr>
<tr>
<td>-s, --security</td>
<td>Show the security service.</td>
</tr>
<tr>
<td>-u, --ui</td>
<td>Show processes belonging to the UI, appserver, and reports.</td>
</tr>
<tr>
<td>-v, --vault</td>
<td>Show the vault service.</td>
</tr>
<tr>
<td>-C, --cmdb</td>
<td>Show CMDB Sync services.</td>
</tr>
</tbody>
</table>

The default is to show all services and ECA engines.

#### User example

In the following example, type the commands on a single line. Line breaks are provided to make the example easier to read.

#### Displaying discovery and reasoning services

The following command displays the processes making up the discovery and reasoning services.

```bash
$TIDEWAY/bin/tw_top --discovery --reasoning
```

**tw_pattern_management**

The **tw_pattern_management** utility enables you to upload pattern modules to the appliance, activate or deactivate pattern modules on the appliance, and remove pattern modules that are no longer required. You can perform these operations on knowledge uploads (Technology Knowledge updates (TKUs)), individual pattern modules, or files.

You can review the functionality provided by the **tw_pattern_management** utility on the **Knowledge management** page. This page enables you to better understand and use the knowledge updates and patterns installed on BMC Atrium Discovery.

- **Pattern** — A sequence of commands written in the Pattern Language (TPL), which contain instructions that identify scanned entities which are then used to create the BMC Atrium Discovery data model.
- **Module** — A text file written in TPL which contains one or more patterns.
- **Knowledge uploads**— Typically these are TKU zip files that include:
  - Zip archives of patterns
• HRD files
• Device RPMs (DRPMS)

Options in the utility specify the actions to be performed on the pattern modules. To use the utility, type the following command:

```
tw_pattern_management [options] <upload/[upload:]module/file>
```

where:

• `upload/[upload:]module/file` is a knowledge upload name, a pattern module identifier, or an existing file. The necessary format to specify a pattern module is `upload name:pattern module name`. You can use shell parameter expansion to upload multiple matching files or all files in the current directory.

• `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--activate-all</code></td>
<td>Activate all pattern modules</td>
</tr>
<tr>
<td><code>--activate-module</code></td>
<td>Activate a specific pattern module</td>
</tr>
<tr>
<td><code>--activate-upload</code></td>
<td>Activate pattern modules within knowledge upload</td>
</tr>
<tr>
<td><code>--deactivate-module</code></td>
<td>Deactivate a specific pattern module</td>
</tr>
<tr>
<td><code>--deactivate-upload</code></td>
<td>Deactivate pattern modules within knowledge upload</td>
</tr>
<tr>
<td><code>-f, --force</code></td>
<td>Deactivate patterns before they are removed</td>
</tr>
<tr>
<td><code>--install</code></td>
<td>Install pattern modules within knowledge upload. It does not activate the pattern modules.</td>
</tr>
<tr>
<td><code>--install-activate</code></td>
<td>Install and activate pattern modules within knowledge upload</td>
</tr>
<tr>
<td><code>-l, --list-uploads</code></td>
<td>List knowledge uploads</td>
</tr>
<tr>
<td><code>--remove-all</code></td>
<td>Remove all pattern modules</td>
</tr>
<tr>
<td><code>--remove-module</code></td>
<td>Remove a specific pattern module</td>
</tr>
<tr>
<td><code>--remove-upload</code></td>
<td>Remove the knowledge upload</td>
</tr>
<tr>
<td><code>--show-progress</code></td>
<td>Display the progress</td>
</tr>
<tr>
<td><code>--upgrade</code></td>
<td>Skip the invalid modules and display compilation messages</td>
</tr>
</tbody>
</table>

✅ Recommendation
To understand TPL patterns and how they might function in your environment, refer to Configipedia, BMC's community website that facilitates knowledge sharing around TPL patterns. Configipedia also provides visibility of the Technology Knowledge Update release schedule and contents.

**Recommendation**

If you build your own zip archives of patterns, make sure they comply with the standard zip format. Other formats will not work.

---

### User examples

In the following examples, type the commands on a single line. Line breaks are provided to make the examples easier to read.

#### Uploading and activating several patterns within a knowledge upload

You can upload and activate several patterns as a knowledge upload by specifying the knowledge upload name and using the `--install-activate` option. To upload and activate several patterns:

1. Change to the directory containing the TPL zip archives.

   ```
   tideway@appliance01 ~] $ cd tmp
   tideway@appliance01 tmp] $ ls
   UploadA.zip  UploadB.zip
   ```

2. Use the utility to upload patterns (providing your password to run the command):

   ```
   tideway@appliance01 tmp] $ tw_pattern_management --install-activate *
   password:
   Uploaded UploadA.zip as knowledge upload "Upload A"
   Uploaded UploadB.zip as knowledge upload "Upload B"
   Knowledge uploads "Upload A", "Upload B" activated
   tideway@appliance01 tmp] $
   ```

#### Deactivating a pattern module

You can deactivate a pattern module by specifying the pattern module name and using the `--deactivate-module` option. To deactivate a pattern module, type the following command:

```
tideway@appliance01 ~] $ tw_pattern_management --deactivate-module "Host.ComputerSystem"
password:
1 pattern module deactivated

tideway@appliance01 ~] $

tw_atssso_control

The `tw_atssso_control` utility enables you to manage BMC Atrium SSO integration for user authentication.

**Recommendation**

Use the BMC Atrium Discovery user interface to perform the functionalities provided by the `tw_atssso_control` command line utility (see Integrating with BMC Atrium Single Sign-On). If there are issues BMC Atrium SSO configuration issues because of which you must disable the integration and login as a local user, you may choose to run the utility.

To use the utility, type the following command:

```
tw_atssso_control [options]
```

where, `options` are any of the options described in the following table and the common command line options described in Using command line utilities.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--deregister</td>
<td>Deregisters the BMC Atrium SSO Web Agent.</td>
</tr>
<tr>
<td>--disable</td>
<td>Disables the BMC Atrium SSO integration.</td>
</tr>
<tr>
<td>--enable</td>
<td>Enables the BMC Atrium SSO integration.</td>
</tr>
<tr>
<td>--force</td>
<td>Force stops potentially destructive actions.</td>
</tr>
<tr>
<td>--fqdn=ARG</td>
<td>Specifies the fully qualified domain name for the BMC Atrium Discover appliance.</td>
</tr>
<tr>
<td>--quiet</td>
<td>Specifies that the user does not receive any informational messages.</td>
</tr>
<tr>
<td>--register</td>
<td>Registers the BMC Atrium SSO Web Agent.</td>
</tr>
<tr>
<td>--sso-password=ARG</td>
<td>Enables you to enter the BMC Atrium SSO administrator password for registration of the web agent.</td>
</tr>
<tr>
<td>--sso-realm=ARG</td>
<td>Specifies the BMC Atrium SSO realm.</td>
</tr>
<tr>
<td>--sso-url=ARG</td>
<td>Specifies the BMC Atrium SSO server URL.</td>
</tr>
<tr>
<td>--sso-username=ARG</td>
<td>Specifies the BMC Atrium SSO administrator username for registration of the web agent.</td>
</tr>
</tbody>
</table>
User example

The following is an example of how to disable BMC Atrium SSO integration by using the utility:

```
$TIDEWAY/bin/tw_atsso_control --disable
```

tw_run_upgrade

In BMC Atrium Discovery 10.0 and later versions, you can perform the following upgrades on standalone machines and clusters either from the BMC Atrium Discovery UI or by running the `tw_run_upgrade` command line utility:

- Upgrade to a BMC Atrium Discovery Service Pack or a version later than 10.0
- Upgrade the Operating System

⚠️ The preferred way to upgrade is through the UI instead of the `tw_run_upgrade` utility. The utility is recommended only if there is an upgrade issue and you are required to resolve it by using the options provided by the utility.

tw_run_upgrade capabilities

The `tw_run_upgrade` utility is an interactive command line tool. Based on the upgrade issue you have, it informs you about the option you must run the utility with and any additional action you are required to perform to resolve it. To learn about some of the typical scenarios where you will use the `tw_run_upgrade` utility, see Resolving an incomplete upgrade process.

Running tw_run_upgrade

Before you start an upgrade, make sure that you have downloaded the compressed upgrade archive from the BMC Electronic Product Distribution (EPD) site and copied that to the `/usr/tideway/var/upgrade` directory of the machine from which you will run the upgrade.

To run the upgrade using the `tw_run_upgrade` utility, you must login to the machine from which you will run the upgrade as the root user and type the following command:

```
tw_run_upgrade [options]
```

where `options` are any of the options described in `tw_run_upgrade options` and the common command line options described in Using command line utilities.
**tw_run_upgrade options**

The following table describes the `tw_run_upgrade` utility options which are not listed in *Using command line utilities*:

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--auto</td>
<td>Specifies to suppress prompts for confirmation of continuing the upgrade after time estimation.</td>
</tr>
<tr>
<td>--distribute</td>
<td>Specifies to prepare the upgrade and distribute the upgrade files over the cluster. It does not run the upgrade process. You can distribute the upgrade files first and run the upgrade later.</td>
</tr>
<tr>
<td>--fix-interrupted</td>
<td>Specifies to unlock the machine after a previous upgrade run was interrupted.</td>
</tr>
<tr>
<td>--no-version-check</td>
<td>Specifies not to check the upgrade compatibility.</td>
</tr>
<tr>
<td>--restart</td>
<td>Specifies to unlock the machine where the upgrade was interrupted and restart the upgrade on that machine.</td>
</tr>
<tr>
<td>--start</td>
<td>Specifies to prepare the upgrade, distribute the upgrade files (if not already distributed), and start the upgrade process. This is the default mode of execution.</td>
</tr>
<tr>
<td>--tmpdir=TMP_DIR_PATH</td>
<td>Specifies to set up the upgrade temporary directory. If a machine is likely to run out of disk space, you can set a symlink for the upgrade directory to a different mounted file system which has free disk space.</td>
</tr>
</tbody>
</table>

**Resolving an incomplete upgrade process**

The following section contains user examples of some typical scenarios where you will use the `tw_run_upgrade` utility in a **standalone machine** and a **cluster**:

**User example for a standalone machine**

If the upgrade process is interrupted on a standalone machine, the command line directs you to run the upgrade process again by running the `tw_run_upgrade --start` command.

**User examples for a cluster**

- Using the `restart` command option: While running the upgrade, if the upgrade process is interrupted on one of the machines in the cluster after the services were stopped, it puts that machine into a locked state and prevents the other members of the cluster (where the upgrade has completed) from starting. If you attempt to start the services manually from the command line of the machine where the upgrade was interrupted, it directs you to unlock it and resume the upgrade process by running the following command:

  ```
  sudo tw_run_upgrade --restart
  ```

Running the command unlocks that machine and resumes the upgrade process. Once the machine is upgraded and it reboots, all the machines in the cluster will also start.
**Warning**

The upgrade must be restarted (``tw_run_upgrade --restart``) only if it fails at a stage during the actual execution of the upgrade script. If the upgrade fails at an earlier stage (for example, during preparing the upgrade or distributing the upgrade files on the cluster), restarting will only upgrade the machine from where you are running the upgrade.

- **Using the ``--fix-interrupted`` command option:** While running the upgrade, if the upgrade process is interrupted on one of the machines in the cluster before the services were stopped, it puts that machine into a locked state and displays an error message on the machine from where you are running the upgrade:

  
  Member no longer aware of ADDM Upgrade operation

  The interruption stops the services only on the machine where the upgrade was interrupted. If you attempt to start the services on that machine manually from the command line, it directs you to unlock it by running the following command:

  ```
  sudo tw_run_upgrade --fix-interrupted
  ```

  Running the command unlocks that machine. The interactive command line tool informs you if any additional intervention is required before you can run the ``--start`` option and run the upgrade again for that machine.

**tw_model_wipe**

The ``tw_model_wipe`` utility enables you to delete all data in the datastore. The utility does not delete configuration data held outside the datastore. However, some configuration information is held in the datastore and is lost when running ``tw_model_wipe``. This is:

- Scan ranges
- Scheduled scan ranges
- Exclude ranges
- CAM (saved queries, groups and subgroups, named values, functional component definitions, and generated patterns)
- DDD Removal blackout windows
- Configuration as a scanner or consolidator
- Saved Queries (on the Reports page)
- CMDB Sync configuration (both connections and filters)

**Warning**
The `tw_model_wipe` utility deletes all data in your datastore. Before you use `tw_model_wipe` to resolve an issue, it is recommended that you contact BMC Customer Support.

To use the utility, type the following command at the `$TIDEWAY/bin/` directory:

```
tw_model_wipe [options]
```

where *options* are any of the options described in the following table and the common command line options described in [Using command line utilities](#).

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--fix-interrupted</code></td>
<td>Unlocks the machine if the previous run of the utility failed to complete.</td>
</tr>
<tr>
<td><code>--force</code></td>
<td>Specifies to remove any existing tideway.db datastore.</td>
</tr>
</tbody>
</table>

**Warning**

Use this argument with caution! It deletes all data stored in your datastore!

---

**Effect on CMDB synchronization**

The `tw_model_wipe` utility deletes the contents of the datastore. When you rescan your estate, the algorithm that identifies hosts (actually root nodes, which are Hosts, Network Devices, MFParts, Printers, and SNMP Managed Devices) may come up with a different identifier or key each time a root node is newly created in the datastore. However, the CMDB relies on the keys of the root nodes remaining the same; if they are not the same, then the CMDB creates duplicate CIs.

To avoid duplicate CI creation in the CMDB, BMC Atrium Discovery version 10 introduces a root node key information export utility which enables you to extract information and the key of root nodes from a machine before reinitialization. You can then upload this to the reinitialized machine where it is stored in the datastore as `RootNodeKeyInfo` nodes.

When you rescan the estate with the reinitialized machine, any potentially new root node is compared against the `RootNodeKeyInfo` nodes, and if a match is found for the discovery target, a new node is created using the key from the `RootNodeKeyInfo` node and the `RootNodeKeyInfo` node is deleted. If no match is found in the `RootNodeKeyInfo` nodes, then a new node is created with a new unique key.

To learn how to use the root node key information export utility, see [preserving IDs when using `tw_model_wipe`].
tw_cluster_control

The `tw_cluster_control` utility enables you to perform the following operations:

- Review the status of all the machines across the cluster.
- Stop the services across the cluster.
- Restart the services across the cluster.
- Remove all failed machines from a cluster.
- Revert a cluster member into a standalone machine.
- Unlock the system when it is locked due to a cluster manager operation failure.
- Change the coordinator when the UI is inaccessible.

To use the utility, type the following command at the `$/TIDEWAY/bin/` directory on a member of the cluster you need to control:

```
tw_cluster_control [options]
```

where:

- **options** are any of the options described in the following table and the common command line options described in **Using command line utilities**.

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>--become-coordinator</code></td>
<td>Make this machine the coordinator.</td>
</tr>
<tr>
<td><code>--cluster-start-services</code></td>
<td>Start the services on all machines across the cluster.</td>
</tr>
<tr>
<td><code>--cluster-stop-services message=MSG</code></td>
<td>Message giving the reason for stopping the services across the cluster. Used in conjunction with <code>--cluster-stop-services</code>.</td>
</tr>
<tr>
<td><code>--cluster-stop-services</code></td>
<td>Stop the services on all machines across the cluster. This option prompts for the password of the system user.</td>
</tr>
<tr>
<td><code>--fix-interrupted</code></td>
<td>Unlock the system when it is locked due to a cluster manager operation failure.</td>
</tr>
<tr>
<td><code>--force</code></td>
<td>Do not ask for confirmation for any of the options.</td>
</tr>
<tr>
<td><code>--replace-vm-uuid</code></td>
<td>Replaces a cluster member's VM UUID if it has changed, preventing the cluster from starting. Use this if the cluster does not start and logs the following critical message:</td>
</tr>
</tbody>
</table>
### Command Line Option Description

<table>
<thead>
<tr>
<th>Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--replace-vm-uuid</td>
<td>Replace expected VM UUID by running: <code>tw_cluster_control --replace-vm-uuid</code> in <code>tw_svc_cluster_manager.log</code>. See Troubleshooting clusters for more information.</td>
</tr>
<tr>
<td>--remove-broken</td>
<td>Remove all failed machines from a cluster. You should use this if you are unable to forcibly remove one or more failed machines using the UI. This option prompts for the password of the system user.</td>
</tr>
<tr>
<td>--revert-to-standalone</td>
<td>Revert the local failed cluster member into a standalone machine. You should only use this after removing a failed machine from the cluster using the --remove-broken option. This option prompts for the password of the system user.</td>
</tr>
<tr>
<td>--show-members</td>
<td>Show the status of all the machines across the cluster.</td>
</tr>
<tr>
<td>--show-pending</td>
<td>Show any pending changes in the cluster.</td>
</tr>
</tbody>
</table>

### User examples

In the following examples, you can stop and restart the services across the cluster, and launch troubleshooting operations if the cluster members are not accessible or are locked by BMC Atrium Discovery.

#### Review the status of the cluster members

You can request information about the current status of all machines in the cluster using the following command:

```
$TIDEWAY/bin/tw_cluster_control --show-members
```

Following are examples of the cluster status details. Click here to expand...

#### Cluster status example for a totally healthy cluster

This example contains the status information for a cluster where all members operate without failures and there are no connectivity issues.

Cluster UUID : d5933b313a3ef13dfe647f00000104a7  
Cluster Name : ADDMCluster  
Cluster Alias :  
Number of Members : 3

UUID : d5933b313a3ef13de8027f00000104a7  
Name : ADDMCluster-01  
Address : 10.49.16.61
Cluster status example with errors
This is an example of the cluster health check results for the case when only cluster coordinator is operating normally and other members of the cluster are down and inaccessible.

<table>
<thead>
<tr>
<th>UUID</th>
<th>Name</th>
<th>Address</th>
<th>Health</th>
<th>State</th>
<th>Coordinator</th>
<th>Last Contact</th>
<th>CPU Type</th>
<th>Processors</th>
<th>Memory</th>
<th>Swap</th>
<th>Free Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>5de1a3313a3f03a67d627f00000104a8</td>
<td>ADDMCluster-02</td>
<td>10.49.17.64</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_STATE_NORMAL</td>
<td>Yes</td>
<td>Thu Nov 28 10:28:20 2013</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
<td>1</td>
<td>384M</td>
<td>8192M</td>
<td>/usr 4955M/8701M (44%)</td>
</tr>
<tr>
<td>2029c0313a3ef7b8af7c7f00000104a5</td>
<td>ADDMCluster-03</td>
<td>10.49.17.67</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_STATE_NORMAL</td>
<td>No</td>
<td>Thu Nov 28 10:28:20 2013</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
<td>1</td>
<td>384M</td>
<td>8192M</td>
<td>/usr 5168M/8701M (41%)</td>
</tr>
<tr>
<td>d5933b313a3ef13de8027f00000104a7</td>
<td>ADDMCluster-01</td>
<td>10.49.16.61</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_STATE_NORMAL</td>
<td>Yes</td>
<td>Thu Nov 28 10:24:46 2013</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
<td>1</td>
<td>384M</td>
<td>8192M</td>
<td>/usr 5170M/8701M (41%)</td>
</tr>
</tbody>
</table>
Memory : 384M
Swap : 8192M
Free Space : /usr 4955M/8701M (44%)

UUID : 5de1a3313a3f03a67d627f00000104a8
Name : ADDMCluster-02
Address : 10.49.17.64
Health : MEMBER_HEALTH_ERROR Communication failure
State : MEMBER_STATE_NORMAL
Coordinator : No
Last Contact : None
CPU Type :
Processors : 0
Memory : 0M
Swap : 0M

UUID : 2029c0313a3ef7b8a1fc7f00000104a5
Name : ADDMCluster-03
Address : 10.49.17.67
Health : MEMBER_HEALTH_ERROR Communication failure
State : MEMBER_STATE_NORMAL
Coordinator : No
Last Contact : None
CPU Type :
Processors : 0
Memory : 0M
Swap : 0M

Restarting the services across the cluster

$TIDEWAY/bin/tw_cluster_control --cluster-start-services

Stopping the services across the cluster

$TIDEWAY/bin/tw_cluster_control --cluster-stop-services
 --cluster-stop-message="Machine is not responding"

Unlock the system when it is locked due to a cluster manager operation failure

Some cluster management operations might acquire the system lock. If the operation is interrupted while the system is in a locked state, you might need to run the following command to unlock it:

$TIDEWAY/bin/tw_cluster_control --fix-interrupted
Running the command unlocks only the machine that was affected by the interrupted operation. The interactive command line tool informs you if any additional intervention is required before you can run the `tw_cluster_control` operations again for that machine.

**Further examples**

For further examples using `tw_cluster_control` to troubleshoot cluster problems, see troubleshooting clusters.
Developing

This section contains information for developers. The following topics are provided:

- **Node lifecycle** — describes the internal components of BMC Atrium Discovery data model and is useful if you plan to extend the data model.
- **The Pattern Language TPL** — a reference guide for developers on the proprietary pattern language that is used for extending the discovery by adding custom patterns.

Node lifecycle

This document is intended for the customers and partners of BMC Software and for the integrations developers. The information contained in this document will help you understand the BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) node lifecycle. This document details the internal components of BMC Atrium Discovery, the data model (for example, how virtualization hosts are modeled), and how to extend the data model. It is a good reference guide for integrating BMC Atrium Discovery with other systems.

The prerequisite for understanding the content in this document is basic familiarity with BMC Atrium Discovery, and the document should be used in conjunction with the Using and Administering sections of the BMC Atrium Discovery documentation.

This document contains the following sections:

- Model summary
- The default data model
- How nodes are identified
- Discovered attributes by platform
- How nodes are removed
- Relationships
- Attachments
- Partitions and history
- Key terminology
- Inferred nodes
- DDD nodes
- Pattern Management nodes
- Conjecture nodes

The nodes are described in terms of their attributes, relationships and lifecycle.

Nodes can also be created by a user; some of their attributes and relationships are only populated by the user through the User Interface. These specific instances are noted in the documentation.
Model summary

The BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) data model and the approach to data storage enables you to model complex IT environments in such a way that the model is a very close representation of the actual environment. This document describes the mapping between the model and the real-world environment in detail. The model is constructed from discovery information, possibly augmented with data imported from other sources, and is an approximation of the actual state of the environment. For more information, see The default data model.

The Datastore

The datastore uses a graph model to represent entities in the environment and the relationships between them. The basic elements of the storage model showing the nodes, roles and relationships are represented in the following example:

This diagram illustrates the graph model of the datastore.

All data is stored as nodes. Nodes have a kind, such as Host, and a set of key-value pair attributes representing the state of the node. Values can be of any type, including complex structures and sequences. However, in the majority of cases, the values are simple scalar types such as strings and numbers, or sequences of scalar types.

Relationships represent connections between nodes, for example, to indicate that a process is running on a particular host computer. Relationships are attached to nodes through Roles. The Roles remove ambiguities in two main situations:

- When both nodes have the same kind: For example, to model an Employment relationship between two Person nodes, one Person would have the role Employer and other would have the role Employee.
- When the same kind of relationship can exist for different reasons: For example, a Person can have the role BusinessOwner or SupportOwner in an Ownership relationship, while many different kinds of node have the OwnedItem role at the other end of the relationship.

The datastore itself does not maintain any knowledge of what kinds of nodes, roles and relationships are expected, or of the keys and value types expected as node and relationship state. These details are maintained by the taxonomy subsystem. The taxonomy defines the expected node, Role and Relationship kinds, and the attributes which they are expected to have. The definitions in the taxonomy determine the model of the environment.
Types of information

The model makes explicit distinctions between the different types of information that are stored. The core types of information are as follows:

- **Inferred** — see Inferred nodes.
- **Directly Discovered** — see DDD nodes.
- **Pattern Management** — see Pattern Management nodes.

BMC Atrium Discovery also stores provenance and imported information.

Inferred information is the type of information that is likely to be of most interest to most users and is inferred from other information using rules in the Reasoning Engine. This type of information is unlikely to change between scans and includes classifications of hosts, groupings of discovered information into running instances of products, and so forth. See Inferred nodes.

In contrast, directly discovered information is obtained directly from a target host through particular discovery techniques. This includes items such as a list of running processes associated with a host, network connections, and so forth. For more information, see DDD nodes.

Imported information is information about discovery targets acquired from a source other than the target itself. For example, a storage management system, or an Active Directory domain manager.

Pattern Management information is any information that is included in a pattern and describes what could be discovered and inferred about an environment, as opposed to what has actually been discovered. The Technology Knowledge Network (TKN) provides some of this information, but it also includes customer-specific knowledge about the applications they use and rules about how they are connected. For more information, see Pattern Management nodes.

Provenance relationships are meta-information describing how the other information came to exist. It is automatically generated as the Reasoning Engine builds and maintains the model. For more information about provenance, see Provenance.

The default data model

The default data model diagrams show the BMC Atrium Discovery default shipping model including the main sections of the model. The diagrams should be useful if you are building searches and using the model to understand it. The aim of the diagrams are to be a quick reference to complement the detail recorded in this Node Lifecycle information and the taxonomy.

It does not represent the entire BMC Atrium Discovery data model and therefore does not show all the nodes, attributes or relationships.

Further details of the data model can be found in the BMC Atrium Discovery taxonomy. To view the taxonomy on the BMC Atrium Discovery UI, from the Administration tab, click View Taxonomy. A link to the current, full-size version of the diagrams below is also available in the online documentation.
Taxonomy limitation

The relationships displayed in the taxonomy are not exhaustive. That is, all possible relationships are not displayed in the taxonomy. For example, the following containment relationships are valid:

- Group:Container:Containment:ContainedItem:SoftwareInstance
- Group:Container:Containment:ContainedItem:Host

The following diagrams show the BMC Atrium Discovery Default Data Model with its different nodes, attributes and relationships between different parts of BMC Atrium Discovery. It is split across the following diagrams:

- Main Diagram—contains all the core entities used in the modeling and discovery of hosts.
- Network and Printer Diagram—contains entities used in the modeling and discovery of network devices and printers.
- Mainframe Diagram—contains entities used in the modeling and discovery of mainframe computers.
- Storage and Load Balancers—contains entities used in the modeling and discovery of storage entities and load balancers.

It is important to note that the model itself is not in separate sections, the separate diagrams exist in order to more clearly convey information.
### Color-Code Key

- Inferred nodes and connection links— **Blue** .
- DDD nodes and connection links— **Green** .
- Knowledge (Pattern Management) nodes and connection links— **Pink** .
- Provenance Relationship connection links— **Red** .
- Auxiliary nodes and connection links—Brown (used internally by BMC Atrium Discovery).

You can also download a pdf file [here](#) which includes all of the diagrams above.

### Node Kinds

#### Inferred nodes

For details on these nodes see [Inferred nodes](#).

- Host node
- Host Container node
- Software Instance node
- Software Component node
- Business Application Instance node
- Functional Component node
- Mainframe node
- MPart node
- Coupling Facility node
- Cluster node
- File node
- File System node
- Package node
- Patch node
- Fibre Channel nodes
• Network Device node
• Printer node
• SNMP Managed Device node
• Management Controller node
• Load Balancer nodes
  • Load Balancer Group node
  • Load Balancer Instance node
  • Load Balancer Member node
  • Load Balancer Pool node
  • Load Balancer Service node
• IP Address node
• Network Interface node
• Subnet node
• Detail node
• Database Detail node
• Storage node
• Storage Collection node
• Generic Element node
• Runtime Environment node
• Storage nodes
  • Storage Device node
  • Storage System node
  • Storage Pool node
  • Storage System Group node
  • Storage Volume node
  • Storage Connection node
  • Storage Processor node
  • Disk Drive node

DDD nodes

For details on these nodes see DDD nodes.

• Introductory information
• Discovery Run node
• Discovery Access node
• Device Info node
• Host Info node
• Directory Listing node
• Registry Listing node
• Service List node
• Discovered Directory Entry node
• Discovered Registry Entry node
• Discovered Service node
• Other DDD nodes
• Integration DDD nodes
  • Integration Point node
  • Integration Result node
  • SQL Result Row node
  • Provider Access node

Pattern Management nodes
For details on these nodes see Pattern Management nodes.

• Pattern node
• Pattern Configuration nodes
• Pattern Define nodes
• Pattern Definitions node
• Pattern Module node
• Knowledge Upload node
• Rule Module node
• Root Node Key Info node

Conjecture nodes
For details on these nodes see Conjecture nodes.

• Automatic Group node

How nodes are identified
Nodes are identified by their key. Keys remain stable because the key is generated from properties that BMC Atrium Discovery has discovered about the real world entity.

The key attribute
All inferred and pattern management node kinds have a key attribute. The key attribute contains the system's best attempt at generating a unique key to identify the real-world entity represented by the node.

The key attribute is necessary to compare and reconcile the data with external systems. The internal node ID cannot be used for this purpose because it is an identifier for the node stored in the datastore, not for the entity it represents. In certain circumstances, the node representing a real-world entity can be removed, and a new one created later. In that situation, the node and its node ID have changed, even though the real-world entity has not changed.

The key attribute is defined to be a text field. The mechanism for generating a key varies for different kinds of nodes, and even for different cases of a particular node kind.
Unique datastore identifier

There are several things that can be unique about nodes. An example of something that is unique is the internal node ID. The node ID is a unique datastore identifier for that node in the datastore. All nodes in the datastore have a unique identifier that is assigned when the node is created.

This is an internal identifier that is used as an index by the database. It can be used to uniquely identify a node, though you should note that it is not persisted on destruction and creation of a new node which represents the corresponding real-world entity.

For this reason, node IDs are not guaranteed to remain stable over time. If you need to identify a node uniquely over time in order to effect an integration with any external system, BMC strongly recommends that you use the key attribute of the node, rather than the node ID.

For details on what makes a host unique, see Host node.

Discovered attributes by platform

An overview of the attributes that BMC Atrium Discovery aims to recover from the different OS platforms is shown in the tables below. The attribute descriptions for each node kind are detailed in the relevant node sections later in this document.

The discovered attributes per OS platform for Inferred nodes is shown in the Table below. An X denotes that the attribute can be discovered. See OS platform key below.

The table shows the attributes from various OS platforms that can be discovered by a default BMC Atrium version 10.2 system. This does not include patterns or a TKU.

Some of my attributes are missing!

Some attributes might not be discovered on a given OS in your environment due to privilege requirements, hardware and software configurations, customizations, firewall configurations, security policies, and so forth. Additional attributes might be discovered by patterns, whether TKU provided, or your own custom patterns.

Click here to download a pdf version of this page.

OS platform key

- A - AIX
- B - FreeBSD
- C - HPUX
- D - IBM i
- E - IRIX
- F - Linux
- G - MPE/iX
- H - Mac OS X
- I - NetBSD
- J - Netware
- K - OpenBSD
- L - OpenVMS
- M - POWER HMC
- N - Solaris
- P - Tru64
- R - UnixWare
- S - VMware ESX
- T - VMware ESXi
- U - Windows

<table>
<thead>
<tr>
<th>Node/Attribute</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
<th>N</th>
<th>P</th>
<th>R</th>
<th>S</th>
<th>T</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores per processor</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cpu threading enabled</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Dns domain</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domain</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Host_type</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hostid</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hostname</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Kernel</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Ldom name</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Ldom role</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Logical Ram</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Lpar partition number</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Num logical processors</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Num processor types</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Num processors</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Os</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Os_arch</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Os_build</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Node/Attribute | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| Node/Attribute      | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| os_class           | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_edition         |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |   |
| os_level           |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |
| os_type            | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_vendor          | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| os_version         | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| processor_speed    | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| processor_type     | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| PSU status         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |
| ram                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| serial             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| service_pack       |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |
| threads_per_core   | X |   |   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| uuid               |   | X |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |
| vendor             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| workgroup          |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |
| zone_uuid          |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |
| zonename           |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |

**Patch**

| Node/Attribute | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| name           | X |   |   |   |   |   |   |   | X | X | X | X | X | X | X | X | X | X | X |

**Package**

| Node/Attribute | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| arch           |   | X | X | X |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| description    | X | X | X |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| epoch          |   | X | X | X |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| name           | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| pkgname        |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| revision       | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| vendor         | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| version        | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

**NetworkInterface**
| Node/Attribute           | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| adapter_type            |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |
| aggregated_by           | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aggregated_with         |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aggregates              | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aggregation_mode        | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| bonded                  |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| default_gateway         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| description             |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| dhcp_enabled            |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| dhcp_server             |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| dns_servers             |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| driver_date             |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| driver_version          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| duplex                  |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| group_name              |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |   |   |   |   |
| ifindex                 |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| interface_type          |   | X |   |   |   |   |   |   |   |   |   |   |   |   | X | X |   |   |   |
| is_console              |   |   |   |   |   |   |   |   |   |   |   |   |   | X | X |   |   |   |   |
| mac_addr                |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| name                    |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

| Node/Attribute           | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| negotiation             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| physical_adapters       |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| physical_location       |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| ppa                     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| primary_wins_server     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| secondary_wins_server   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| service_name            |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| setting_id              |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| shared_adapters         |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| speed                   |   | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| virtual_adapters        |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

| Node/Attribute           | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
|                         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| Node/Attribute         | A | B | C | D | E | F | G | H | I | J | K | L | M | N | P | R | S | T | U |
| **IP Address**         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| broadcast             | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| fqdn                  | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| ip_addr               | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| netmask               | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| prefix                | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| prefix_length         | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X|
| temporary             |   |   |   |   |   |   |   |   | X | X | X | X | X |   |   |   |   |   |   |
| zone                  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   | X |
| **FibreChannelHBA**   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aix_manufacturer_code | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| aix_part_number       | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| boot_bios             | X |   |   |   |   |   |   | X | X |   |   |   |   |   |   |   |   |   |   |
| driver_name           | X |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| driver_version        | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| firmware              | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| hba_id                | X | X | X | X |   |   |   |   |   | X | X | X | X |   |   |   |   |   |   |
| model                 |   |   |   |   |   |   |   |   |   | X | X | X | X |   |   |   |   |   |   |
| option_rom_version    | X | X |   |   | X | X | X | X |   |   |   |   |   |   |   |   |   |   |   |
| optrom_bios_version   | X | X |   | X |   |   |   |   | X |   |   |   |   |   |   |   |   |   |   |
| optrom_efi_version    | X | X |   | X |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |
| optrom_fcode_version  | X | X |   | X |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |
| optrom_fw_version     | X |   |   | X |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |
| serial                |   | X | X | X |   |   |   |   |   |   |   |   | X |   |   |   |   |   |   |
| **FibreChannelNode**  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| firmware              | X | X | X |   | X | X | X | X |   |   |   |   |   |   |   |   |   |   |   |
| wwnn                  | X | X | X | X |   | X | X | X |   |   |   |   |   |   |   |   |   |   |   |
| **FibreChannelPort**  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| fabric_name           |   | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| port_state            | X | X | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
Restricted PDFs

This topic describes and links to PDFs and other documents that support this product release. If the ready-made PDFs of this space do not satisfy your requirements, you can export a custom PDF.

Tip

You can create a custom PDF for a specific topic, a topic and its children, or a group of topics with a specific label. For information about how you can export a custom PDF from this space, see Exporting to PDF and other formats.

Related topics

Where to find more information

FAQs and additional resources

Ready-made PDFs of this space

Writer: Create and post PDFs of your complete space when the space is first released (GA) and for every subsequent release vehicle (service pack, patch, maintenance release, and so on) that is documented in this space. Having these PDFs available for download will divert PDF-seeking users from using the PDF export functionality, which stresses the system. Follow these guidelines for the PDFs page:

- Include all customer-facing content in the PDF. Omit the PDFs and Restricted PDFs topics, the Tracking tools topic, and all topics outside of the Home page hierarchy, such as:
  - Legal notices
  - Third-party product terms of use
  - Inclusions libraries
Providing a space PDF at SMA release is not required. If you do this, you must later replace the SMA PDF with the GA space PDF to simplify the experience for most of your users. Be sure to update the document properties in the Ready-made PDFs table.

Do not create PDFs for subsets of the space. Creating PDFs of selected topics and sections will fragment the documentation into guides again, introduce inconsistency among products, and require extra production work (for example, different cover pages for each PDF).

If your space was released to customers before April 19, 2012, export a PDF of your current content and then for every future release vehicle. In this situation, name your first PDF for your most recent release vehicle and, if you made updates to the space after the release, include the Post-prefix. The example entries in the table below list PDFs for a space released well before the space PDF requirement, where Service Pack 4 came out earlier in 2012, and the IDD team made changes to the space between that service pack and the first snapshot.

After you attach the PDFs and other supporting documents to this topic, populate the tables with the title (not file name), document creation date, file size, and description of each document. Link to each attachment from its title in the tables. When adding entries to the table, ensure that the table is sorted by date, with the most recent document in the first row.

The following table provides ready-made PDFs that contain snapshots of the content in this space. Ready-made PDFs are created for new releases and service packs at the time of their release. Although these PDFs contain all topics, the content of some topics is better suited for online viewing.

### Ready-made PDFs

<table>
<thead>
<tr>
<th>Snapshot</th>
<th>Date</th>
<th>File size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version 7.6.03</td>
<td>July 15, 2012</td>
<td>300 KB</td>
</tr>
<tr>
<td>Version 7.6.02 SP5</td>
<td>May 13, 2012</td>
<td>300 KB</td>
</tr>
<tr>
<td>Post-Version 7.6.02 SP4</td>
<td>April 19, 2012</td>
<td>298 KB</td>
</tr>
</tbody>
</table>

### Product guides

The following table lists PDFs of product guides that are not available as topics in the documentation portal.

### Product guide PDFs

<table>
<thead>
<tr>
<th>Title</th>
<th>Date</th>
<th>File size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>docTitleAsLink</td>
<td>creationDate</td>
<td>fileSize KB</td>
<td>Describes...</td>
</tr>
</tbody>
</table>
How nodes are removed

It is important that the model of your network environment stays current over time and that it is maintained efficiently. This not only involves creating new nodes to represent recently discovered data, but also the removal of existing data that is no longer present in your environment. BMC Atrium Discovery views the environment so that different kinds of nodes are removed in different ways. Different methods are used to remove inferred nodes and Directly Discovered Data nodes from the model. These variations in behavior are detailed in the following sections.

Inferred nodes removal

The following sections detail the ways in which inferred nodes are removed from BMC Atrium Discovery. For more information on inferred nodes, see Inferred nodes.

Manual

To remove a node manually, select it in the node list and pick Destroy from the Actions list.

Authoritative

BMC Atrium Discovery has inferred the presence of a node from data such as an interface list, where there is clear evidence that the node exists. When BMC Atrium Discovery cannot see any evidence of this node, it no longer exists. Therefore, BMC Atrium Discovery removes the node automatically from the model. Aging is not applied to this type of removal.

Non-authoritative (aging)

BMC Atrium Discovery has inferred the presence of a node from data such as a process list, where there is clear evidence that the node exists. However, in cases when BMC Atrium Discovery cannot see any evidence that the node still exists, it does not necessarily imply that the node no longer exists. There might be other reasons why the node appears to no longer exist.

For example, a process that relates to a software instance (SI), such as an Adobe FrameMaker desktop application, might appear to be running during an initial scan, but not on a subsequent scan. This does not necessarily indicate that Adobe FrameMaker no longer exists; instead, it is possible that the user has stopped running the application during the subsequent scan. Because it is the nature of IT infrastructure to have frequent minor changes to configurations, hosts, and software, discovered data becomes less relevant over time.

Aging is applied to this type of removal to prevent frequent removal and recreating of SIs. When the node has been in the aging state for some time, it is then removed from the model.

Only the following classes of nodes age:
Root nodes (Host, Network Device, MFPart, Printer, SNMP Managed Device, and Management Controller.)

First order Software Instances (ones created directly from DDD)

Runtime Environment

Storage

The Software Instance, Runtime Environment, and Storage nodes mentioned above are only automatically aged if they are triggered on certain types of DDD. These are described on the individual node pages.

Although there are important considerations before doing so, you can modify data aging limits in the Model Maintenance settings of the user interface.

Secondary authoritative

This type of removal is authoritative, however it is based on different criteria than the original evidence for its existence.

An example of this would be a batch process, where a node creation (and therefore existence) is triggered from seeing the job running in the process list, but BMC Atrium Discovery might check the timestamp of a log file to see if it believes it is running regularly.

This type will apply to any removal which involves actively seeking evidence about whether something exists.

Containment

This type of removal applies when BMC Atrium Discovery has inferred nodes which have a containment relationship to another node. These are generally nodes which model physical items such as network interface cards. If the non-authoritative removal process (Aging) has caused BMC Atrium Discovery to remove a host, then the network interfaces that are contained in that host will also cease to exist. Therefore, Aging is not applied and the inferred node is removed.

Cascade

This type of removal applies where further inferred nodes are constructed from first order or other inferred nodes. Any aging or specific removal will already have taken place at the first order nodes, so there is no need to apply any further removal strategies. In this case the removal should cascade up.

Directly Discovered Data nodes removal

The removal process for DDD nodes, is a much simpler process than it is for Inferred nodes.

The five removal strategies for Inferred nodes calculate that the entity in the real-world to which the data corresponds no longer exists and therefore the node is removed. Each time you run a discovery of DDD nodes, a new set of data is retrieved. If the data is older than a certain cut-off point and BMC Atrium Discovery discovers more recent data which refers to the same node, the node is removed.
Removal groups in the pattern language

When patterns construct complicated structures, such as DatabaseDetail nodes when doing deep discovery of databases, removal must be performed by the pattern that creates them. Removal Groups are named groups of nodes which simplify deletion of these complex structures. They are described in Model functions in the TPL Guide.

Relationships

Node associations are represented by relationships. There are different types of relationships which define hierarchy, hosting, containment, ownership, manufacturer, category, versions, location, dependency, in fact, any kind of relationship required to model an IT infrastructure. A type of relationship is represented as colon-separated strings. A relationship comprises more than the two node endpoints, it contains the "route" between the two. This is made up in the following way:

```
Node:Role:RelationshipLink:Role:Node
```

For example the relationship from a Host node to the Cluster node that contains the host is shown as:

```
Host:ContainedHost:HostContainment:HostContainer:Cluster
```

The relationship from a Cluster node to a Host node that it contains is shown as:

```
Cluster:HostContainer:HostContainment:ContainedHost:Host
```

⚠️ A relationship is a bidirectional link between two nodes. In this section, the terms "start node" and "target node" are used to describe the way that the relationship is built. However, once the relationship has been built, the terms are redundant.

The general timeline for the creation of a relationship is:

1. Create "start node".
2. Create "target node".
3. Create relationship between 1 and 2.

The individual components of the relationship are described in the following sections.
Node

The node is any kind of node that can have a relationship, such as a Host node. All nodes can have relationships, including relationships.

Relationship Kinds

The relationship kind describes the nature of the relationship. For example, the DeviceSubnet relationship kind describes the link between a device and the subnet that it is on, but without any directionality implied.

⚠️ This section refers to Relationship Kinds as the colon-separated lists which describe how two nodes are related, the roles that each node plays in the relationship, and the Relationship Link which describes the nature of the Relationship.

A complete list of Relationship Kinds with their descriptions is provided in the table below.

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy Relationships</td>
<td></td>
</tr>
<tr>
<td>Hierarchy</td>
<td>Relates nodes into a hierarchy.</td>
</tr>
<tr>
<td>Attachment Relationships</td>
<td></td>
</tr>
<tr>
<td>Attachment</td>
<td>Relates a node to a file attachment.</td>
</tr>
<tr>
<td>Category Relationships</td>
<td></td>
</tr>
<tr>
<td>ElementCategory</td>
<td>Relates an element to a category it is in.</td>
</tr>
<tr>
<td>People Relationships</td>
<td></td>
</tr>
<tr>
<td>Ownership</td>
<td>Relates an element to its owner.</td>
</tr>
<tr>
<td>Management</td>
<td>Relates an employee to their manager.</td>
</tr>
<tr>
<td>Locations</td>
<td></td>
</tr>
<tr>
<td>Location</td>
<td>Relates an element to the location it is in.</td>
</tr>
<tr>
<td>LocationContainment</td>
<td>Relates a location to the larger location it is in.</td>
</tr>
<tr>
<td>Hosts</td>
<td></td>
</tr>
<tr>
<td>HostedSoftware</td>
<td>Relates software to the host it is running on.</td>
</tr>
<tr>
<td>HostContainment</td>
<td>Relates hosts and software instances to contained virtual hosts.</td>
</tr>
<tr>
<td>EndpointIdentity</td>
<td>Relates Hosts that appear to have changed identity</td>
</tr>
<tr>
<td>Applications</td>
<td></td>
</tr>
<tr>
<td>SoftwareContainment</td>
<td>Relates an item of software to an item contained in it.</td>
</tr>
<tr>
<td>SoftwareService</td>
<td>Relates an item of software to a service.</td>
</tr>
<tr>
<td>Communication</td>
<td>Relates software instances to communicating software instances.</td>
</tr>
<tr>
<td>Relationship Kind</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>Dependency</td>
<td>Relates elements to elements they depend upon.</td>
</tr>
<tr>
<td>Physical Model</td>
<td></td>
</tr>
<tr>
<td>DeviceSubnet</td>
<td>Relates an IP device to the subnet it is on.</td>
</tr>
<tr>
<td>DeviceInterface</td>
<td>Relates an IP device with one of its network interfaces.</td>
</tr>
<tr>
<td>NetworkLink</td>
<td>Relates a network interface on a host to a port interface on a switch.</td>
</tr>
<tr>
<td>HostedFile</td>
<td>Relates a file to its host.</td>
</tr>
<tr>
<td>RelatedFile</td>
<td>Relates a file to something that uses it.</td>
</tr>
<tr>
<td>FibreChannelNodeDevice</td>
<td>Relates a World Wide Node Number (WWNN) to an HBA.</td>
</tr>
<tr>
<td>FibreChannelNodePort</td>
<td>Relates a World Wide Port Number (WWPN) to a WWNN.</td>
</tr>
<tr>
<td>FileSystemMount</td>
<td>Relates the mounter and the FileSystem.</td>
</tr>
<tr>
<td>Directly Discovered Data</td>
<td></td>
</tr>
<tr>
<td>DiscoveryAccessResult</td>
<td>Relates a DiscoveryAccess to one of its results.</td>
</tr>
<tr>
<td>List</td>
<td>Relates a List of Members.</td>
</tr>
<tr>
<td>Sequential</td>
<td>Relates sequential nodes.</td>
</tr>
<tr>
<td>Inference</td>
<td>Relates an inferred node with its source node.</td>
</tr>
<tr>
<td>Status</td>
<td>Relationship to a status node.</td>
</tr>
<tr>
<td>AccessFailure</td>
<td>Access failure.</td>
</tr>
<tr>
<td>AccessOptimization</td>
<td>Access optimization.</td>
</tr>
<tr>
<td>Metadata</td>
<td>Information about a discovery request.</td>
</tr>
<tr>
<td>Pattern Relationships</td>
<td></td>
</tr>
<tr>
<td>PatternModuleContainment</td>
<td>Associates a pattern module to its contents.</td>
</tr>
<tr>
<td>PatternModule</td>
<td>Associates a pattern module to its pattern package.</td>
</tr>
<tr>
<td>PatternModuleDependency</td>
<td>Associates a pattern module to its dependents.</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Pattern that is maintaining an inferred node.</td>
</tr>
<tr>
<td>Requirement</td>
<td>Associates a pattern that will need the products of another.</td>
</tr>
<tr>
<td>Override</td>
<td>Associate patterns that are being overridden or are overriding others.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>Associated pattern has been deprecated or is deprecating others.</td>
</tr>
<tr>
<td>Error</td>
<td>Errors generated by this pattern at runtime.</td>
</tr>
<tr>
<td>PatternExecution</td>
<td>Associates a PatternExecution to its Pattern.</td>
</tr>
<tr>
<td>PatternTouch</td>
<td>Associate a PatternTrigger with things it changed.</td>
</tr>
<tr>
<td>PatternTrigger</td>
<td>Associates a PatternExecution to its trigger nodes.</td>
</tr>
<tr>
<td>Request</td>
<td>Associates a Pattern to the DDD nodes requested by it.</td>
</tr>
<tr>
<td>IntegrationImplementation</td>
<td>Relates the implementation of a resource.</td>
</tr>
<tr>
<td>Relationship Kind</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>ResourceUse</td>
<td>Relates the usage of a resource</td>
</tr>
</tbody>
</table>

**Generic Modeling Relationships**

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containment</td>
<td>Relates an element to its container.</td>
</tr>
<tr>
<td>Detail</td>
<td>Relates an element to details about it.</td>
</tr>
<tr>
<td>SupportDetail</td>
<td>Relates Host and Software Instance nodes to SupportDetail nodes.</td>
</tr>
</tbody>
</table>

**DIP Relationships**

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provides</td>
<td>Relates an IntegrationProvider to all its created IntegrationPoints.</td>
</tr>
<tr>
<td>IntegrationPointContainment</td>
<td>Containment of one node by another</td>
</tr>
<tr>
<td>ConnectionUsage</td>
<td>Relates an IntegrationResult to the connection that was used to generate that result</td>
</tr>
<tr>
<td>QueryUsage</td>
<td>Relates an IntegrationResult to the query that was used to generate that result</td>
</tr>
</tbody>
</table>

**Rule Relationships**

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RuleModuleDependency</td>
<td>Associates a rule module to its dependents.</td>
</tr>
<tr>
<td>EnablingRange</td>
<td>Associates a discovery access with an IP range for revisiting purposes.</td>
</tr>
</tbody>
</table>

**Foundation User Relationships**

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Favorite</td>
<td>A Favorite item for a User</td>
</tr>
<tr>
<td>InvokingUser</td>
<td>Relates to the invoking user</td>
</tr>
</tbody>
</table>

**Hardware Reference Data Relationships**

<table>
<thead>
<tr>
<th>Relationship Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReferenceData</td>
<td>Relates physical hosts to hardware reference data</td>
</tr>
</tbody>
</table>

**Role**

The role describes the part that its node plays in the relationship.

For example, where a Host node represents a host which is running a software process:

- A Software Instance has been created which represents the process running on the host.
- The Host node is acting in the role of Host; the host for some running software.
- The Software Instance is acting in the role of RunningSoftware; software which is running on a host.

Roles are required because a node can play one of many parts in a relationship, and clarification is needed; the role clarifies the part that the nodes are playing in the relationship.

A complete alphabetical list of Role Kinds in the default taxonomy with their descriptions is provided in the table below.

<table>
<thead>
<tr>
<th>Role Kind</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Associate</td>
<td>Associate inference.</td>
</tr>
<tr>
<td>Role Kind</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>Attachment</td>
<td>Attachment.</td>
</tr>
<tr>
<td>AttachmentContainer</td>
<td>Node containing an attachment.</td>
</tr>
<tr>
<td>BusinessOwner</td>
<td>Business owner of an element.</td>
</tr>
<tr>
<td>Category</td>
<td>Category of elements.</td>
</tr>
<tr>
<td>Child</td>
<td>Child in hierarchy.</td>
</tr>
<tr>
<td>Client</td>
<td>Client.</td>
</tr>
<tr>
<td>CouplingFacility</td>
<td>Coupling Facility.</td>
</tr>
<tr>
<td>Contained</td>
<td>An element contained within another.</td>
</tr>
<tr>
<td>ContainedHost</td>
<td>Host contained within another host.</td>
</tr>
<tr>
<td>ContainedLocation</td>
<td>A location inside another location.</td>
</tr>
<tr>
<td>ContainedSoftware</td>
<td>Piece of software contained inside other software.</td>
</tr>
<tr>
<td>Container</td>
<td>An element containing others.</td>
</tr>
<tr>
<td>Contributor</td>
<td>Contributor inference.</td>
</tr>
<tr>
<td>Dependant</td>
<td>Entity that depends upon another.</td>
</tr>
<tr>
<td>DependedUpon</td>
<td>Entity depended upon by another.</td>
</tr>
<tr>
<td>Detail</td>
<td>A detail belonging to an element.</td>
</tr>
<tr>
<td>DeviceOnSubnet</td>
<td>IP device belonging to a subnet.</td>
</tr>
<tr>
<td>DeviceWithInterface</td>
<td>A device with an interface, for example, a network interface.</td>
</tr>
<tr>
<td>DeviceWithAddress</td>
<td>A device with an IP address.</td>
</tr>
<tr>
<td>DiscoveryAccess</td>
<td>Discovery access.</td>
</tr>
<tr>
<td>DiscoveryResult</td>
<td>Discovery result.</td>
</tr>
<tr>
<td>DiscoveryRun</td>
<td>Discovery run.</td>
</tr>
<tr>
<td>EdgeClient</td>
<td>Edge client.</td>
</tr>
<tr>
<td>EdgeDevice</td>
<td>Edge device.</td>
</tr>
<tr>
<td>Element</td>
<td>Element.</td>
</tr>
<tr>
<td>ElementInCategory</td>
<td>Element belonging to a category.</td>
</tr>
<tr>
<td>ElementInLocation</td>
<td>Element in a location.</td>
</tr>
<tr>
<td>ElementWithDetail</td>
<td>Element with associated details.</td>
</tr>
<tr>
<td>Collection</td>
<td>Element containing a collection of elements.</td>
</tr>
<tr>
<td>ElementWithStatus</td>
<td>Element with an associated status.</td>
</tr>
<tr>
<td>ManagedElement</td>
<td>Element being managed.</td>
</tr>
<tr>
<td>Employee</td>
<td>Employee.</td>
</tr>
<tr>
<td>Role Kind</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>EndpointRange</td>
<td>A range used to control endpoint access.</td>
</tr>
<tr>
<td>FibreChannelDeviceWithNode</td>
<td>A Fibre Channel HBA with a WWNN.</td>
</tr>
<tr>
<td>FibreChannelNode</td>
<td>A Fibre Channel WWNN for a HBA.</td>
</tr>
<tr>
<td>FibreChannelNodeWithPort</td>
<td>A Fibre Channel WWNN with a WWPN.</td>
</tr>
<tr>
<td>FibreChannelPort</td>
<td>A Fibre Channel WWPN.</td>
</tr>
<tr>
<td>File</td>
<td>A File.</td>
</tr>
<tr>
<td>Hardware</td>
<td>Physical host to which hardware reference data will be related.</td>
</tr>
<tr>
<td>HardwareDetail</td>
<td>Support Detail Data is for Hardware.</td>
</tr>
<tr>
<td>HomeLocation</td>
<td>Home location of an element.</td>
</tr>
<tr>
<td>Host</td>
<td>Host for Software Instances.</td>
</tr>
<tr>
<td>HostContainer</td>
<td>Host containing other hosts.</td>
</tr>
<tr>
<td>HostedFile</td>
<td>A File on a Host.</td>
</tr>
<tr>
<td>ITOwner</td>
<td>IT owner of an element.</td>
</tr>
<tr>
<td>InferredElement</td>
<td>Inferred element.</td>
</tr>
<tr>
<td>InferredRelationship</td>
<td>Inferred relationship.</td>
</tr>
<tr>
<td>InstalledSoftware</td>
<td>Installed software.</td>
</tr>
<tr>
<td>InterfaceOfDevice</td>
<td>Interface of an IP device.</td>
</tr>
<tr>
<td>InterfaceWithAddress</td>
<td>Interface with an IP address.</td>
</tr>
<tr>
<td>IPv4Address</td>
<td>An IPv4 address.</td>
</tr>
<tr>
<td>IPv6Address</td>
<td>An IPv6 address.</td>
</tr>
<tr>
<td>KnowledgeUpload</td>
<td>Knowledge Upload.</td>
</tr>
<tr>
<td>List</td>
<td>List.</td>
</tr>
<tr>
<td>Location</td>
<td>Current location of an element.</td>
</tr>
<tr>
<td>LocationContainer</td>
<td>A location containing other locations.</td>
</tr>
<tr>
<td>Manager</td>
<td>Manager</td>
</tr>
<tr>
<td>ManufacturedItem</td>
<td>Manufactured item.</td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Manufacturer of an item.</td>
</tr>
<tr>
<td>Member</td>
<td>Member of a list or collection.</td>
</tr>
<tr>
<td>MountedFileSystem</td>
<td>Mounted file system.</td>
</tr>
<tr>
<td>Mounter</td>
<td>Mounter of a file system.</td>
</tr>
<tr>
<td>NetworkEndpoint</td>
<td>One end of a network connection.</td>
</tr>
<tr>
<td>New</td>
<td>A new node.</td>
</tr>
<tr>
<td>Old</td>
<td>A replaced node.</td>
</tr>
<tr>
<td>Role Kind</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>Next</td>
<td>Next node in sequence.</td>
</tr>
<tr>
<td>OperatingSystem</td>
<td>Operating system of a host.</td>
</tr>
<tr>
<td>OSDetail</td>
<td>Support Detail Data is for Operating System.</td>
</tr>
<tr>
<td>OwnedItem</td>
<td>Item owned by someone or something.</td>
</tr>
<tr>
<td>Owner</td>
<td>Owner of an item.</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent in hierarchy.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Pattern.</td>
</tr>
<tr>
<td>PatternConfiguration</td>
<td>Pattern configuration.</td>
</tr>
<tr>
<td>PatternDefine</td>
<td>Pattern definitions function.</td>
</tr>
<tr>
<td>PatternDefinitions</td>
<td>Pattern definitions.</td>
</tr>
<tr>
<td>PatternModule</td>
<td>Pattern module.</td>
</tr>
<tr>
<td>PatternExecution</td>
<td>Pattern execution.</td>
</tr>
<tr>
<td>PatternTrigger</td>
<td>Pattern trigger.</td>
</tr>
<tr>
<td>PatternCreated</td>
<td>Created node.</td>
</tr>
<tr>
<td>PatternModified</td>
<td>Modified node.</td>
</tr>
<tr>
<td>PatternConfirmed</td>
<td>Confirmed node.</td>
</tr>
<tr>
<td>PatternDestroyed</td>
<td>Destroyed node.</td>
</tr>
<tr>
<td>Peer</td>
<td>Peer.</td>
</tr>
<tr>
<td>Previous</td>
<td>Previous node in sequence.</td>
</tr>
<tr>
<td>Primary</td>
<td>Primary inference.</td>
</tr>
<tr>
<td>ReferenceData</td>
<td>Holds hardware reference data.</td>
</tr>
<tr>
<td>Relationship</td>
<td>Relationship.</td>
</tr>
<tr>
<td>Resource</td>
<td>Resource being used.</td>
</tr>
<tr>
<td>ResourceUser</td>
<td>User of resource.</td>
</tr>
<tr>
<td>RunningSoftware</td>
<td>Software running on a host.</td>
</tr>
<tr>
<td>AggregateSoftware</td>
<td>Aggregate software running on a host.</td>
</tr>
<tr>
<td>Service</td>
<td>Service being provided.</td>
</tr>
<tr>
<td>ServiceProvider</td>
<td>Provider of a service.</td>
</tr>
<tr>
<td>Server</td>
<td>Server.</td>
</tr>
<tr>
<td>SoftwareContainer</td>
<td>Piece of software containing other software.</td>
</tr>
<tr>
<td>SoftwareDetail</td>
<td>Support Detail Data is for Software.</td>
</tr>
<tr>
<td>Status</td>
<td>Status.</td>
</tr>
<tr>
<td>StorageContainer</td>
<td>Element containing storage.</td>
</tr>
<tr>
<td>Role Kind</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>ContainedStorage</td>
<td>Storage contained in an element.</td>
</tr>
<tr>
<td>Storage</td>
<td>Storage for an element.</td>
</tr>
<tr>
<td>Subnet</td>
<td>Subnet of an IP device.</td>
</tr>
<tr>
<td>SupportOwner</td>
<td>Person or group responsible for supporting an element.</td>
</tr>
<tr>
<td>Overridden</td>
<td>A pattern that is overridden by another.</td>
</tr>
<tr>
<td>Overrider</td>
<td>A pattern that is overriding another.</td>
</tr>
<tr>
<td>Error</td>
<td>An ECAError related to a pattern.</td>
</tr>
<tr>
<td>PatternWithError</td>
<td>A pattern with ECAErrors.</td>
</tr>
</tbody>
</table>

**Provenance**

By linking data items and their attributes to the evidence for them, BMC Atrium Discovery enables its data to be easily verified; a prerequisite for trusting it. This feature is called *Provenance*.

Provenance Data is created by the Reasoning Engine and consists of relationships between Directly Discovered Data items and entities inferred from them - a relationship back to the source from the inferred entity. For example, there is a provenance relationship between each Software Instance (SI) and the data that caused the SI to be inferred - typically one or more processes, files, or the like.

Provenance information is meta-information describing how the other information came to exist. It is generated as Reasoning builds and maintains the model. Provenance information is stored on relationships in the model.

- **Inference Types**
  - **Primary** — indicate that the existence of the evidence node is the reason that the inferred node was created.
  - **Contributor** — indicate that the evidence node provided information used in building the inferred node.
  - **Associate** — indicate that BMC Atrium Discovery knows there is a relationship between the evidence node and the inferred node.
  - **Relationship** — indicate that BMC Atrium Discovery knows that a relationship exists because of the evidence node.
  - **Deletion** — indicate that the removal of the inferred node was due to withdrawal of the evidence node.
- **Maintaining Pattern** — the pattern maintaining a node.

If you need to access the data in your model you should access the UI functions provided in BMC Atrium Discovery that enable this. Apart from specialist purposes, you should not need to understand the provenance implementation details in BMC Atrium Discovery. However, an overview is provided below.
Provenance Implementation Details

To represent provenance information, attributes can be stored on the Inference relationships. The keys attribute contains a list of the attribute keys that were set or confirmed on the inferred node as a result of the inference. The rule attribute contains the name of the reasoning rule that decided upon the inference. Provenance information is only created and is not updated.

Inferred nodes are related to DDD nodes using ‘Creation Inference’ relationships. The Inferred node has the role InferredElement; the DDD node has the role Primary, Contributor or Associate as appropriate. To support the full lifecycle of inferred nodes there are complimentary Removal Inference relationships. Destruction is the complimentary role to Primary, and Withdrawal is the complimentary role to Contributor.

The method of linking of discovered information to inferred information means that in general, each inferred node will have relationships to many discovery data nodes, representing multiple discoveries of the entity over time.

The Reasoning Engine maintains this provenance information and this can be accessed using functions in the BMC Atrium Discovery Search Service named Provenance and Provenance Details.

Relationships between Relationships

Relationships are technically a sub-class of nodes and they have the same basic characteristics. Relationships can have attributes stored in them and can have relationships to other nodes in the same way that nodes can.

A relationship connects two nodes between their role, however a relationship can also act in a role in a relationship. Therefore, you can have relationships between nodes and relationships or relationships between relationships.

Relationships between Inferred nodes

Using patterns, the reasoning engine can construct a number of relationships between inferred nodes, as follows:

Links an SI to the Host it is running on.

```
SoftwareInstance:HostedSoftware:HostedSoftware:Host:Host
```

A client-server relationship between SIs. This can be inferred based on the presence of network connections between processes. It can also be based on other evidence such as configuration files or directly querying the endpoints.

```
SoftwareInstance:Client:Communication:Server:SoftwareInstance
```
Equivalent to the client-server case, but where the endpoints in a communication are peers, rather than client and server.

SoftwareInstance:Peer:Communication:Peer:SoftwareInstance

A dependency between SIs, other than a communication link. Attributes on the relationship can indicate the kind of dependency.

SoftwareInstance:Dependant:Dependency:DependedUpon:SoftwareInstance

Represents SIs that logically contain other SIs.

SoftwareInstance:SoftwareContainer:SoftwareContainment:ContainedSoftware:SoftwareInstance

Links a virtual host to the Software Instance that implements it.

SoftwareInstance:HostContainer:HostContainment:ContainedHost:Host

Links a configuration file to a Software Instance.


Represents the SIs that make up a BAI.

BusinessApplicationInstance:SoftwareContainer:SoftwareContainment:ContainedSoftware:SoftwareInstance

Represents BAIs that logically contain other BAIs.

A dependency between BAIs.

BusinessApplicationInstance:Dependant:Dependency:DependedUpon:BusinessApplicationInstance

Indicates the Hosts which form a cluster.

Cluster:HostContainer:HostContainment:ContainedHost:Host

Indicates the SIs that provide clustering services for a cluster.

Cluster:Service:SoftwareService:ServiceProvider:SoftwareInstance

Represents the relationship between logical hosts and their container for Sun E15Ks and similar.

HostContainer:HostContainer:HostContainment:ContainedHost:Host

Represents the relationship between network interfaces and the subnets they are connected to.

Subnet:Subnet:DeviceSubnet:DeviceOnSubnet:NetworkInterface

In addition to these relationships, the reasoning engine also constructs relationships to the Pattern Management nodes, for example, patterns.

Attachments

Attachments are files that can be associated with many different kinds of nodes in the datastore, but they are usually used with hosts. A typical use of attachments is for disaster recovery purposes; all necessary documents can be stored and managed by BMC Atrium Discovery. Any number of attachments can be associated with a single node, and the same file can be attached to multiple nodes. A copy of the attached file is uploaded to the BMC Atrium Discovery appliance for secure storage. If the same file is attached to different nodes, multiple copies are held by BMC Atrium Discovery.

When a node with attachments is marked as destroyed, the attachments are not destroyed, and remain attached to the destroyed node.
Partitions and history

This topic is included for users who are interested in the internal mechanisms used by BMC Atrium Discovery to store data.

The datastore is divided into the following default partitions:

- **Audit** — used by the auditing system. No user interaction is permitted.
- **Conjecture** — used by the Host Grouping system. User interaction is permitted.
- **DDD** — used for Directly Discovered Data in the model. User interaction is permitted.
- **Default** — the default partition for storing the modelled IT infrastructure. User interaction is permitted.
- **Taxonomy** — used to store the taxonomy which is the template defining the nodes, attributes, and relationships used to model networks and applications in the datastore. The taxonomy does not limit you to predefined data, additional kinds can be created too. No user interaction is permitted.
- **_System** — used for system tasks. No user interaction is permitted.

The Default and Taxonomy partitions store a history of all changes for all nodes by default. The Default and Taxonomy partitions can be purged by a user with the `model/datastore/partition/partitionname/write` permission (where `partitionname` is a named partition or `*` for all).

History is not recorded for attributes whose names begin with a double underscore, for example `__patch_checksum`.

Some default reports rely on history being enabled.

When any changes are made to a node, they are recorded by default. The changes can be made by any of the following:

- Automatic changes made by:
  - Reasoning
  - Import tools
- Changes made through the UI

Changes are stored on the node that has been changed. The following items are stored regarding the change:

- **Date Time** — A timestamp for the change.
- **User** — The user who made the change. Internal users, such as Reasoning, are recorded as such.
Key terminology

This document assumes a familiarity with BMC Atrium Discovery and as a reminder here are some of the key terminology and concepts that are used throughout this guide:

- **Data Aging** - Discovered data is regarded as valid at the time of its last successful scan. The nature of IT infrastructure means that frequent, minor changes to configurations, hosts, and software are common. Consequently, discovered data can be regarded as becoming less current with the passing of time. In BMC Atrium Discovery when data passes a certain configurable aging threshold, it is destroyed.

- **Datastore** - All data used by the BMC Atrium Discovery system is held in an object database. The datastore treats data as a set of objects and the relationships between them.

- **Directly Discovered Data** - Data that the Discovery Engine has discovered; it has not undergone any processing beyond simple parsing. Everything that the Discovery Engine finds that might be of interest is stored, regardless of whether it is understood or not. It is stored in a structured form that can be queried and reported on, making it easy to construct certain kinds of discovery reports, and aids in developing new patterns.

- **Event** - The Rules Engine (ECA Engine) executes rules in response to Events.

- **Host** - A node in the model which represents a physical or virtual computer system including information about its OS and its physical or virtual hardware. A host is sometimes referred to as an OSI (Operating System Instance). See Host node.

- **ID** - The ID of a node is a unique identifier for the node itself. If the node corresponding to an entity is destroyed, and a new node is subsequently created for it, the new node will have a different ID, but will have the same key.

- **Inferencing** - This is the act of drawing conclusions based on other data.

- **Key** - The key of a node is a unique identifier for the entity that the node represents. However, the key of a node is persistent unlike the node ID.

- **Lifecycle** - The lifecycle of an entity describes the conditions under which it comes into existence, and the conditions under which it ceases to exist. For nodes in the BMC Atrium Discovery model, the lifecycle stages are:
  - **Current** - The lifecycle stage used to describe nodes which exist in your model. BMC Atrium Discovery has evidence that they currently exist in your environment.
  - **Aging** - The lifecycle stage used to describe nodes which also exist in your model. However, these nodes represent entities that BMC Atrium Discovery has not seen in a certain period of time and has chosen to 'age out' of the model. Not all entities that BMC Atrium Discovery fails to see evidence for after a certain period of time are aged.
  - **Destroyed** - The lifecycle stage used to describe nodes which have been specifically marked as destroyed. These nodes are 'destroyed', however, they remain in the model.
  - **Purged** - The lifecycle stage used to describe destroyed nodes which have been specifically 'purged' from the model. Purging a node means that it no longer exists in the model and has been actually removed from the datastore.
- **Node** - A node is an object in the BMC Atrium Discovery datastore, which represents an entity in the environment. Nodes have a kind, such as 'Host', and a number of named attributes. Nodes can be connected to other nodes via Relationships. Most node kinds have a 'key' which uniquely identifies the entity in the environment.

- **Node Kind** - The type of a node, such as Host or Software Instance. The default set of nodes and their associated attributes and relationships are defined in the BMC Atrium Discovery taxonomy.

- **Pattern** - Patterns are written in the Pattern Language (TPL). Patterns are responsible for creating and maintaining the model. Each pattern in TPL has a corresponding Pattern node in the model, which is related to the nodes that the pattern is maintaining. Patterns are used to extend the functionality of the reasoning engine.

- **Provenance** - Meta-information describing how inferred information came to exist. It is generated as Reasoning builds and maintains the model. Provenance information is stored as relationships in the model.

- **Reasoning Engine** - The Reasoning Engine is an event based engine which orchestrates and drives the population of the different parts of the data model through execution of a series of rules that make up the core functionality of the product. It is extensible through the use of patterns.

- **Relationship** - The way in which two or more nodes are associated with each other. Like nodes, relationships have a kind, such as 'Dependency' and a number of named attributes. Relationships are linked to nodes via Roles.

- **Removal** - The concept of taking data out of the model using one or more of the BMC Atrium Discovery lifecycle methodologies (Aging, Destroyed or Purged). Removal is the collective term used in this document.

- **Role** - A node with a relationship to another node acts in a Role in the relationship, which indicates which end of the relationship it is. For example, in a 'Dependency' relationship, one node has the Role 'Dependant' and the other has the Role 'DependedUpon'.

- **Rules Engine** - This is another term used to describe the reasoning engine. The Rules Engine processes the rules that are generated from Patterns, in order to maintain the model. The Rules Engine is an Event Condition Action (ECA) engine.

- **Rules** - Rules are small fragments of executable code that run in the Rules Engine. Rules are generated from Patterns when they are activated. Other core rules are distributed with BMC Atrium Discovery.

- **Taxonomy** - The template defining the nodes, attributes, and relationships used by BMC Atrium Discovery and stored in the datastore. The BMC Atrium Discovery taxonomy also defines how much of the data model is represented in the user interface.

- **Trigger** - The Trigger for a pattern describes the conditions under which the pattern executes. Triggers correspond to the creation, modification or destruction of node. For further information on the BMC Atrium Discovery terminology, see the Glossary.
Inferred nodes

This section describes the BMC Atrium Discovery Inferred nodes. Inferred information is the type of information that is likely to be of most interest to most users and is inferred from other information using rules in the Reasoning Engine.

- Host node
- Host Container node
- Software Instance node
- Software Component node
- Business Application Instance node
- Functional Component node
- Mainframe node
- MFPart node
- Coupling Facility node
- Cluster node
- File node
- File System node
- Package node
- Patch node
- Fibre Channel nodes
- Network Device node
- Printer node
- SNMP Managed Device node
- Management Controller node
- Load Balancer nodes
  - Load Balancer Group node
  - Load Balancer Instance node
  - Load Balancer Member node
  - Load Balancer Pool node
  - Load Balancer Service node
- IP Address node
- Network Interface node
- Subnet node
- Detail node
- Database Detail node
- Storage node
- Storage Collection node
- Generic Element node
- Runtime Environment node
- Storage nodes
  - Storage Device node
  - Storage System node
Host node

A Host node represents a general purpose computer on the network; any physical or virtual computer system in your organization. A host is of a specific type, such as a server or a desktop which BMC Atrium Discovery gains high quality interactive access to. A host is not an IP-connected device, such as a switch or a router.

The attributes of a Host node contain stable information about the host, such as processor and memory details, its primary host name, OS information, whether it is virtual, and so forth. Hosts are considered inferred information because rules inside the reasoning engine are used to determine which hosts exist based on the DiscoveryAccess results obtained. Hosts will only be created for discovery targets that should be considered hosts, as opposed to switches, printers or other infrastructure.

Host node lifecycle

The following section describes the scenarios in which a Host node is created, updated or destroyed automatically.

Creation/update

In order to maintain the model of hosts in the environment, BMC Atrium Discovery identifies hosts using an internal weighting algorithm, see Uniquely Identifying a Host for more information. Hosts are identified by a number of criteria using this algorithm. The creation, update and removal of a Host node are based on these identities.

A Host node is created in the datastore when a new host is inferred from a successful discovery. A Host is only created when good quality access to the target computer has been established. Good quality access is necessary for the BMC Atrium Discovery internal system to establish that there is sufficient stable information available to identify a Host. Stable information is considered to be data such as hardware details, how many active interfaces there are, bios serial numbers and so forth.

A Host node is not created unless information is available from the getDeviceInfo, getHostInfo, and getMACAddresses methods.

For as long as BMC Atrium Discovery's host identification algorithm can determine that a particular discovery target corresponds to a particular Host node, the Host node is updated every time the target is scanned on any of its IP addresses. Once a Host node is created, BMC Atrium Discovery will update its attributes as the host changes over time, while still retaining the original Host node and key.
When a Host node is first created, BMC Atrium Discovery constructs a key attribute that uniquely identifies the host. The key is based on a combination of attribute values from the Host, but those attributes can change over time. As the host is rescanned, the Host node attributes that contributed to the key can all change, but the key itself will not be changed. If a Host node is destroyed and then the corresponding host is rescanned, a new Host node will be created. If the attribute values that were originally used to construct the key have not changed, the new Host node will be given the same key as the original one, but if any of the attributes have changed, the new Host node will have a different key to the old one.

**Removal**

Host node can be destroyed either manually or automatically.

To remove a host node manually, find the host, select it in the list and pick **Destroy** from the **Actions** list.

By default, a Host node will be automatically destroyed, if BOTH of the following circumstances are fulfilled:

- No update has been made in ten days from any of the IP addresses known for this host.
- No update has been made in the last seven scans on the IP address currently being scanned.

Thus, if a host is not successfully scanned, by default, the Host node representing it will remain in the model for at least 10 days. In this context, "seen" means anything other than no update. The table below shows the default removal criteria for the Host node.

<table>
<thead>
<tr>
<th>Node Kind</th>
<th>Days</th>
<th>Scans</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>10</td>
<td>7</td>
</tr>
</tbody>
</table>

These are the default settings for the number of days and number of scans. You can change these settings on the Model Maintenance page. See **Configuring model maintenance settings** for more information.

⚠️ When a Host node is destroyed, its related nodes are destroyed too. That is, if a Host node is destroyed, all of its software instances, network interfaces, files, and HBAs are also destroyed.

When a node with attachments is destroyed, the attachments are not destroyed, and remain attached to the destroyed node. See **Attachments** for an introduction to attachments. If a host node is destroyed from the datastore, all of its relationships are destroyed as well. For example, a host might have a relationship to a location. If the host is destroyed, the relationship to that location will also be destroyed. See **How nodes are removed.**
Uniquely Identifying a Host

There is no single identifying attribute that can be used reliably to identify hosts. BMC Atrium Discovery determines host identities using an internal host identification algorithm to select which existing Host node should be connected, if any, with the Directly Discovered Data from the current scan.

A Host node is only created or updated when BMC Atrium Discovery has enough information to determine that a suitable Host node does not already exist or that an existing Host node is the one to use. A host will not be created or updated unless BMC Atrium Discovery has working credentials to extract enough information from the host.

Once BMC Atrium Discovery has sufficient information on the scanned endpoint it searches in the datastore for candidate Host nodes to be updated. This is carried out using a series of search queries, for example:

- Host previously updated via a scan from this endpoint.
- Host with one of its IP addresses being the endpoint scanned.
- Host with the same bios serial number as the scanned endpoint.

The Host nodes returned by these, and other searches, are candidates for being linked with the scanned Discovery Access node.

All the candidates are then ranked using the BMC Atrium Discovery algorithm based on the data on the candidate and the data currently retrieved from the endpoint. Each attribute on the host could be one of the following:

- Confirmed - same attribute key, same value.
- Changed - the value on the host for that attribute key is different from the value scanned.
- Absent - the value is set on the candidate, but is no longer available in the scanned data.

For each attribute a score is set for confirmed, changed or absent values. These scores are then added together to build the ranking. The score for all the attributes are then added together.

Once all the candidates have been evaluated, Host nodes with a negative ranking are ignored, and the highest positive-ranked Host node will be chosen. If no Host nodes have a positive score, or if no candidates have been found, a new Host node is created. In all cases, the Host node is then updated with the new data.

Endpoint identity

BMC Atrium Discovery scans IP addresses, or endpoints. It is the job of the Host identification algorithm to match endpoints to Host nodes. When an endpoint is scanned, the host identification algorithm sometimes decides that the endpoint now corresponds to a different Host node to the previous scan of that endpoint. This can happen for three main reasons:

1. IP addresses have been reassigned so the endpoint really does correspond to a different host.
2. The endpoint still corresponds to the same host, but the host has changed so much since
   the previous scan that BMC Atrium Discovery cannot tell that it is the same one.
3. The endpoint still corresponds to the same host, but BMC Atrium Discovery's access
   credentials have changed substantially, meaning the data appears very different. This can
   happen when switching between SNMP and login access, for example.

All three situations look the same from BMC Atrium Discovery’s point of view — the discovered
data is significantly different from the data previously retrieved from the endpoint. To facilitate
understanding of these scenarios, BMC Atrium Discovery creates an EndpointIdentity
relationship between the Host nodes. The Host view displays the relationship and allows
comparison of the new and old Host nodes.

In all situations, BMC Atrium discovery will ultimately resolve the condition. If the change is due
to an IP address reassignment, the original Host can be found by scanning a different endpoint. If that
happens, the Host is connected to the new endpoint and the EndpointIdentity relationship is
removed. In other situations, the original Host node will age out according to the normal ageing
rules.

**Virtualization and Consolidation**

BMC Atrium Discovery performs consolidation of virtual and partitioned hosts. When it does this,
the model it constructs is as follows:

- Virtual Hosts - see Virtual Hosts.
  - Contained Hosts - see Contained Hosts.
  - Clustered Hosts - see Clustered Hosts.

**Virtual Hosts**

Virtual Hosts are modeled so that the virtual host is related through the software instance to the
physical host. For example:

- VMware Virtual Machines
- Solaris Containers

![Diagram showing HostedSoftware and HostContainment relationships in virtualized hosting.](image)
Contained Hosts

Where a number of IP devices are discovered that are logical devices on hosts, they are modeled as shown in the diagram below. See Host node lifecycle.

For example, the following types of servers are treated in this manner:

- Sun Enterprise 10000 - E10K
- Sun Fire 12000 - F12K
- Sun Fire 15000 - F15K
- Sun Fire 20000 - F20K
- Sun Fire 25000 - F25K

Clustered Hosts

Where a number of collaborative hosts are discovered (they are seen as individual IP devices and there is no parent), they are consolidated into clusters which are modeled as Cluster nodes, see Cluster node lifecycle. For example, the following types of cluster are treated in this manner:

- Microsoft Clusters
- Veritas Cluster Servers
Graphic showing HostContainment relationships and the relationships between the clustering service SI and the contained hosts.

**Host node attributes**

The attributes and relationships of a Host node are described in the tables below. Where hidden attributes are included, they are described in the most relevant section. Not all attributes are discovered for every platform, see the Discovered attributes by platform page for information on this.

Fields that are displayed in the user interface but have no related attribute on a Host node, that is, they are inherited or populated by relationships, are also shown.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General Details</strong></td>
<td></td>
</tr>
<tr>
<td>name string</td>
<td>The name that the host is known by, typically the same as hostname.</td>
</tr>
<tr>
<td>hostname string</td>
<td>The name that the host is known by systems external to this host.</td>
</tr>
<tr>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>short_name</td>
<td>Short name.</td>
</tr>
<tr>
<td>description string</td>
<td>Description of the host. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>url string</td>
<td>URL for information about the host. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td></td>
<td>If true, the host is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>business_continuity_critical</td>
<td>If true, the host is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>third_party_support boolean</td>
<td>Other names by which this host is known. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>host_type boolean</td>
<td>Type such as ‘UNIX Server’, ‘Windows Desktop’, and so on. This attribute is deprecated, use type instead.</td>
</tr>
<tr>
<td>type string</td>
<td>Type such as ‘UNIX Server’, ‘Windows Desktop’, and so on.</td>
</tr>
<tr>
<td>vendor string</td>
<td>The hardware vendor, for example, Dell Computer Corporation.</td>
</tr>
<tr>
<td>virtual boolean</td>
<td>Flag if the host is virtual.</td>
</tr>
<tr>
<td>partition boolean</td>
<td>Flag if the host is a partition.</td>
</tr>
<tr>
<td>vm_class_string_</td>
<td>Class of virtual machine.</td>
</tr>
<tr>
<td>ssphostname string</td>
<td>The E10K SSP Hostname, the physical host for a Sun Microsystems “Blade” Servers. This is available on Sun Enterprise 10000 systems only.</td>
</tr>
<tr>
<td>sunfire_domain string</td>
<td>F15K SunFire domain. Common to all blades in a SunFire server. The attribute is set for all Host nodes which are members of the server.</td>
</tr>
<tr>
<td>zonename string</td>
<td>The Solaris Zonename of this host and its container. Currently only Solaris 10 supports zonenames.</td>
</tr>
<tr>
<td>ldom_name string</td>
<td>Solaris LDOM name.</td>
</tr>
<tr>
<td>ldom_role string</td>
<td>Solaris LDOM role.</td>
</tr>
<tr>
<td>lpar_name string</td>
<td>IBM Power LPAR Name.</td>
</tr>
<tr>
<td>lpar_partition_number : int</td>
<td>IBM Power LPAR Number.</td>
</tr>
<tr>
<td>local_fqdn string</td>
<td>The fully qualified domain name of the host.</td>
</tr>
<tr>
<td>dns_domain string</td>
<td>The DNS domain to which the host belongs.</td>
</tr>
<tr>
<td>domain string</td>
<td>The NIS/Windows domain to which the host belongs.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
</tr>
<tr>
<td><strong>Windows Workgroup</strong></td>
<td>workgroup string</td>
</tr>
<tr>
<td><strong>Operating System</strong></td>
<td></td>
</tr>
<tr>
<td>OS</td>
<td>os string</td>
</tr>
<tr>
<td>OS Class</td>
<td>os_class string</td>
</tr>
<tr>
<td>OS Type</td>
<td>os_type string</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version string</td>
</tr>
<tr>
<td>OS Architecture</td>
<td>os_arch string</td>
</tr>
<tr>
<td>OS Update Level</td>
<td>os_level string</td>
</tr>
<tr>
<td>OS Build</td>
<td>os_build string</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>os_vendor string</td>
</tr>
<tr>
<td>Service Pack</td>
<td>service_pack int</td>
</tr>
<tr>
<td>Kernel</td>
<td>kernel string</td>
</tr>
<tr>
<td><strong>Patches and Packages</strong></td>
<td></td>
</tr>
<tr>
<td>Package Count</td>
<td>package_count int</td>
</tr>
<tr>
<td>Patch Count</td>
<td>patch_count int</td>
</tr>
<tr>
<td><strong>Attributes used internally to short circuit some processing</strong></td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__patch_checksum string</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__package_checksum string</td>
</tr>
<tr>
<td><strong>Host aging attributes</strong></td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>age_count int</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_success date</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_failure date</td>
</tr>
<tr>
<td><strong>Hardware and Network</strong></td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
</tr>
<tr>
<td>Serial Number</td>
<td>serial string</td>
</tr>
<tr>
<td>UUID</td>
<td>uuid string</td>
</tr>
<tr>
<td>Unique Host Id</td>
<td>hostid string</td>
</tr>
<tr>
<td>Logical RAM</td>
<td>logical_ram int</td>
</tr>
<tr>
<td>Physical RAM</td>
<td>ram int</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>num_processors int</td>
</tr>
<tr>
<td>Processor Type</td>
<td>processor_type string</td>
</tr>
<tr>
<td>Processor Speed</td>
<td>processor_speed int</td>
</tr>
<tr>
<td>Number of Logical Processors</td>
<td>num_logical_processors int</td>
</tr>
<tr>
<td>Cores per Processor</td>
<td>cores_per_processor int</td>
</tr>
<tr>
<td>Threads per Processor Core</td>
<td>threads_per_core int</td>
</tr>
<tr>
<td>CPU Threading Enabled</td>
<td>cpu_threading_enabled boolean</td>
</tr>
<tr>
<td>Number of Processor Types</td>
<td>num_processor_types int</td>
</tr>
<tr>
<td>All Processor Types</td>
<td>_all_processor_types list: string</td>
</tr>
<tr>
<td>All Processor Speeds</td>
<td>_all_processor_speeds list:int</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Logical Processor Counts</td>
<td>_all_processor_logical_counts list:int</td>
</tr>
<tr>
<td>Physical Processor Counts</td>
<td>_all_processor_physical_counts list:int</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>_all_threads_per_core list:int</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>_all_cores_per_processor list:int</td>
</tr>
<tr>
<td>Power Supply Status</td>
<td>psu_status list:string</td>
</tr>
</tbody>
</table>

**Attributes used only to facilitate search queries**

| Not displayed in UI | __all_ip_addrs list:string | BMC Atrium Discovery internal use only. Do not use. Internal attribute to aid searching Hosts by IP address. |
| Not displayed in UI | __all_dns_names list:string | BMC Atrium Discovery internal use only. Do not use. Internal attribute to aid searching Hosts by name. |

**Attributes used for grouping**

| _excluded_from_grouping boolean | BMC Atrium Discovery internal use only. Records that a node has been marked as excluded from automatic grouping. Do not use. |

**Attributes used for CMDB Sync**

| CDM Virtual System Type | cdm_virtual_system_type int | Attribute for populating the CDM BMC_ComputerSystem.VirtualSystemType attribute. |
| Last successful CMDB sync | last_cmdb_sync_success date | The time at which this Host was last successfully synchronized into the CMDB. |
| Last failed CMDB sync | last_cmdb_sync_failure date | The time at which an attempt to synchronize this Host into the CMDB failed. |
| CMDB sync duration | last_cmdb_sync_duration float | The time in seconds spent performing the last CMDB synchronization of this Host. |
| CMDB CI count | last_cmdb_sync_ci_count int | The number of CIs corresponding to this Host at the last CMDB synchronization. |
| CMDB Relationship count | last_cmdb_sync_rel_count int | The number of relationships between CIs corresponding to this Host at the last CMDB synchronization. |

**Host node relationships**

The relationships on a Host node are described in the table below.
<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Containing Group</td>
<td>Host: ContainedItem: Containment: Container: Group</td>
<td>Owning Group.</td>
</tr>
<tr>
<td>Location</td>
<td>Host: ElementInLocation: Location: Location</td>
<td>Location of this Host.</td>
</tr>
<tr>
<td>Hosted Applications</td>
<td>Host: Host: HostedSoftware: AggregateSoftware: BusinessApplicationInstance</td>
<td>Business Application Instances running on this Host.</td>
</tr>
<tr>
<td>Software Instances</td>
<td>Host: Host: HostedSoftware: RunningSoftware: SoftwareInstance</td>
<td>Software Instances running on this Host.</td>
</tr>
<tr>
<td>Aggregate Software Instances</td>
<td>Host: Host: HostedSoftware: AggregateSoftware: SoftwareInstance</td>
<td>Aggregate Software Instances running on this Host.</td>
</tr>
<tr>
<td>Runtime Environments</td>
<td>Host: Host: HostedSoftware: RunningSoftware: RuntimeEnvironment</td>
<td>Runtime Environments on this Host.</td>
</tr>
<tr>
<td>Host Container</td>
<td>Host: ContainedHost: HostContainment: HostContainer: HostContainer</td>
<td>Host container of which this is a member.</td>
</tr>
<tr>
<td>Cluster</td>
<td>Host: ContainedHost: HostContainment: HostContainer: Cluster</td>
<td>Cluster of which this is a member.</td>
</tr>
<tr>
<td>Containing VM</td>
<td>Host: ContainedHost: HostContainment: HostContainer: SoftwareInstance</td>
<td>Software container for this virtual Host.</td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>Host: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: NetworkInterface</td>
<td>Network interfaces of this Host.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>Host: DeviceWithAddress: DeviceAddress: IPv4Address: IPAddress</td>
<td>IPv4 addresses of this Host.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>Host: DeviceWithAddress: DeviceAddress: IPv6Address: IPAddress</td>
<td>IPv6 addresses of this Host.</td>
</tr>
<tr>
<td>Fibre Channel HBAs</td>
<td>Host: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: FibreChannelHBA</td>
<td>Fibre Channel HBAs of this Host.</td>
</tr>
<tr>
<td>Files</td>
<td>Host: Host: HostedFile: HostedFile: File</td>
<td>Files on this Host.</td>
</tr>
<tr>
<td>Device Info</td>
<td>Host: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this Host.</td>
</tr>
<tr>
<td>Host Info</td>
<td>Host: InferredElement: Inference: Primary: HostInfo</td>
<td>HostInfo of this Host.</td>
</tr>
<tr>
<td>Discovery Access</td>
<td>Host: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this Host.</td>
</tr>
<tr>
<td>Patches</td>
<td>Host: Host: HostedSoftware: InstalledSoftware: Patch</td>
<td>Patches on this Host.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------------------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Host: InferredElement: Inference:</td>
<td>Patches contributing to this Host's attributes.</td>
</tr>
<tr>
<td></td>
<td>Contributor: DiscoveredPatches</td>
<td></td>
</tr>
<tr>
<td>Packages</td>
<td>Host: HostedSoftware:</td>
<td>Packages on this Host.</td>
</tr>
<tr>
<td></td>
<td>InstalledSoftware: Package</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Host: InferredElement: Inference:</td>
<td>Packages contributing to this Host's attributes.</td>
</tr>
<tr>
<td></td>
<td>Contributor: DiscoveredPackages</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MountedFileSystem: FileSystem</td>
<td></td>
</tr>
<tr>
<td>Disk Drives</td>
<td>Host: Container: Containment:</td>
<td>DiskDrives contained in this Host.</td>
</tr>
<tr>
<td></td>
<td>ContainedDrive: DiskDrive</td>
<td></td>
</tr>
<tr>
<td>Disk Drives</td>
<td>Host: Consumer: StorageUse: Provider:</td>
<td>DiskDrives used by this Host.</td>
</tr>
<tr>
<td></td>
<td>DiskDrive</td>
<td></td>
</tr>
<tr>
<td>Storage Volumes</td>
<td>Host: Consumer: StorageUse: Provider:</td>
<td>Storage Volumes used by this Host.</td>
</tr>
<tr>
<td></td>
<td>StorageVolume</td>
<td></td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: Host</td>
<td>Next Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Next Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: Printer</td>
<td>Next Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: ManagementController</td>
<td>Next Management Controller Identity.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Next SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Host: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Host: Previous: EndpointIdentity: Next: StorageDevice</td>
<td>Next Storage Device Identity.</td>
</tr>
<tr>
<td>Chosen Endpoint</td>
<td>Host: Device: ChosenEndpoint: Endpoint: Endpoint</td>
<td>Endpoint used to discover this Host.</td>
</tr>
<tr>
<td>Hardware Reference Data</td>
<td>Host: Hardware: ReferenceData: ReferenceData: HardwareReferenceData</td>
<td>Hardware Reference data corresponding to this Host.</td>
</tr>
<tr>
<td>Hardware Support Details</td>
<td>Host: ElementWithDetail: SupportDetail: HardwareDetail: SupportDetail</td>
<td>Hardware Support Detail related to this Host.</td>
</tr>
<tr>
<td>OS Support Details</td>
<td>Host: ElementWithDetail: SupportDetail: OSDetail: SupportDetail</td>
<td>OS Support Detail related to this Host.</td>
</tr>
<tr>
<td>Details</td>
<td>Host: ElementWithDetail: Detail: Detail:</td>
<td>Details of this Host.</td>
</tr>
<tr>
<td>Collections</td>
<td>Host: Member: Collection: Collection: Collection:</td>
<td>A collection which this Host is a member of.</td>
</tr>
<tr>
<td>Communicating With</td>
<td>Host: Host: ObservedCommunication: Host: Host</td>
<td>Other Hosts that this Host is communicating with. Only created when automatic grouping is enabled.</td>
</tr>
<tr>
<td>Managed by Software Instance</td>
<td>Host: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software Instances that manage this Host.</td>
</tr>
<tr>
<td>Managed by SNMP Managed Device</td>
<td>Host: ManagedElement: Management: Manager: SNMPManagedDevice</td>
<td>SNMP Managed Device that manages this Host.</td>
</tr>
<tr>
<td>Managed by Management Controller</td>
<td>Host: ManagedElement: Management: Manager: ManagementController</td>
<td>Management Controller that manages this Host.</td>
</tr>
<tr>
<td>Discovery Conditions</td>
<td>Host: ElementWithCondition: DiscoveryCondition: DiscoveryCondition: DiscoveryCondition</td>
<td>Discovery conditions to which this element pertains.</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>Host: ServiceHost: SoftwareService: Service: LoadBalancerMember</td>
<td>Load balancer member.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Host: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
</tbody>
</table>
Host Container node

A Host Container node is used to model a system that acts as a container for hosts but is not a host itself. For example, a blade enclosure which is discovered via a management controller, or a hardware container for logical hosts such as a Oracle (Sun) StarFire system.

Blade enclosures are typically hardware containers which contain a number of physical hosts, a management controller, and various network modules such as network and/or SAN switches.

In Oracle (Sun) StarFire systems, a single hardware device is partitioned into a number of logical hosts. The existence of the hardware device can be inferred from the details of the logical hosts, but it is never directly seen as a network connected device.

Host Container nodes always have a relationship to the Pattern node corresponding to their maintaining pattern.

Host Container Lifecycle

The following section describes the scenarios in which a Host Container is created, updated or removed.

Creation (blade enclosure)

When a chassis Management controller is discovered, this is modeled as a Management Controller node. The creation of a Management Controller node triggers an additional pattern which creates the Host Container node and collects additional information on the contents of the enclosure, and triggers further patterns to relate the components (physical hosts, a management controller, and various network modules such as network and/or SAN switches).
Some host containers, such as the Sun Blade 6000 system, may need to be scanned twice before all relationships are created.

**Creation (physical container of logical hosts)**

Host Containers of this type are created when a pattern discovers conditions which enable it to detect a contained host. Once found, the pattern creates the Host Container that it needs. A relationship link is automatically created to the Host node and the Host Container will have a key generated. The generated key for a Host Container node is entirely dependent on the type of Host Container.

**Update (blade enclosure)**

When a Management Controller node is updated, the related Host Container is updated accordingly.

**Update (physical container of logical hosts)**

The update procedure for a Host Container is also under full control of patterns. When the pattern has identified a contained host, a key is generated for the Host Container. If a Host Container with that key already exists, the Host Container is updated accordingly.

**Removal (blade enclosure)**

A Host Container node is removed when it no longer contains any hosts and it no longer has an associated Network Device or management controller.

**Removal (physical container of logical hosts)**

The default destruction of a Host Container is not under the control of patterns. A Host Container node is removed when the last remaining contained Host node is unlinked and there is no longer any evidence for that host or any of its relationships, for example, to network devices, management controllers or any other root node kind. This is a Cascade Removal type, see [Cascade Removal](#).

**Host Container node attributes**

The attributes and relationships of a Host Container node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of host container.</td>
</tr>
<tr>
<td>Hardware Vendor</td>
<td>vendor string</td>
<td>Hardware vendor of the Host Container.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Hardware model.</td>
</tr>
</tbody>
</table>

Serial number of the Host Container.
### UI Name and Attribute Name and Type

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number serial string</td>
<td>Not displayed in UI</td>
</tr>
<tr>
<td>Not displayed in UI hostid_range list:string</td>
<td></td>
</tr>
<tr>
<td>System Identifier systemid string</td>
<td>IBM Power Systems System Identifier.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>asset_tag string</td>
</tr>
<tr>
<td>Key key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name name string</td>
<td>The name that the Host Container is known by.</td>
</tr>
<tr>
<td>Short Name short_name string</td>
<td>The short name.</td>
</tr>
<tr>
<td>Description description string</td>
<td>Description of the Host Container. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>URL url string</td>
<td>URL for information about the Host Container. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Business Continuity Critical business_continuity_critical boolean</td>
<td>If true, the Host Container is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Supported by 3rd Party third_party_support boolean</td>
<td>If true, the Host Container is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Synonyms synonyms string</td>
<td>Other names by which this Host Container is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
</tbody>
</table>

### Host Container node relationships

The relationships on a Host Container node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host Members</td>
<td>HostContainer: HostContainer: HostContainment: Host</td>
<td>Hosts that are within this container.</td>
</tr>
<tr>
<td>Network Device Members</td>
<td>HostContainer: HostContainer: HostContainment: Contained: NetworkDevice</td>
<td>Network Devices that are within this container.</td>
</tr>
<tr>
<td>SNMP Managed Device Members</td>
<td>HostContainer: HostContainer: HostContainment: Contained: SNMPManagedDevice</td>
<td>SNMP Managed Devices that are within this container.</td>
</tr>
<tr>
<td>Management Controller Members</td>
<td>HostContainer: HostContainer: HostContainment: ContainedHost: ManagementController</td>
<td>Management Controllers that are within this container.</td>
</tr>
<tr>
<td>Managed by Software Instance</td>
<td>HostContainer: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software Instances that manage this container.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Managed by SNMP Managed Device</td>
<td>HostContainer: ManagedElement: Management: Manager: SNMPManagedDevice</td>
<td>SNMP Managed Device that manages this container.</td>
</tr>
<tr>
<td>Managed by Management Controller</td>
<td>HostContainer: ManagedElement: Management: Manager: ManagementController</td>
<td>Management Controller that manages this container.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>HostContainer: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this container.</td>
</tr>
<tr>
<td>Details</td>
<td>HostContainer: ElementWithDetail: Detail: Detail</td>
<td>Details for this container.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>HostContainer: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>HostContainer: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>HostContainer: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Family</td>
<td>HostContainer: ElementInCategory: ElementCategory: Category: Family</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>HostContainer: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>HostContainer: ElementInLocation: Location: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>HostContainer: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>HostContainer: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>HostContainer: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

**Software Instance node**

A Software Instance (SI) node represents an instance of an off-the-shelf software product or equivalent proprietary item of software. It can correspond to one single running process or a group of processes (possibly on multiple hosts). A Software Instance often corresponds to a licensable entity. Where possible, versions of Software Instances are retrieved and stored. They are created, maintained and destroyed by patterns. Software Instance nodes always have a relationship to the Pattern node corresponding to their maintaining pattern.

- A first order Software Instance is one which models a piece of software running directly on the OS.
- A second order Software Instance is one which model software formed from first order Software Instances.
**Why an SI and not a Runtime Environment?**

Software Instances represent pieces of software running on a host. A Runtime Environment represents something supporting running software. For example, Weblogic is represented by an SI, and Java by a Runtime Environment node. However, it is still important to keep track of the supporting Runtime Environments and their versions as some might require updates.

**Software Instance Lifecycle**

The lifecycle of a Software Instance node depends upon the details of the pattern that maintains it. This is determined by a combination of the pattern trigger condition, whether the pattern specifies an explicit key for the Software Instance and the logic contained in the pattern itself. In all cases, if a pattern is deleted, the Software Instance nodes it is maintaining are immediately destroyed (as are all other nodes it might be maintaining).

Patterns can trigger on a Directly Discovered Data node, see [Directly Discovered Data Trigger](#). They can also trigger on the creation or modification of other Software Instance nodes, see [Software Instance Trigger](#).

**Directly Discovered Data Trigger**

**Creation/update**

If a pattern triggers on a Directly Discovered Data node, such as a Discovered Process node or a Discovered Listening Port node, it might choose whether to specify keys for the Software Instance nodes it creates and maintains. If a key is specified then the decision whether to create a new Software Instance node or to update an existing one depends on the key. If a Software Instance node with the specified key exists, that node is updated, even if the node was previously maintained by a different pattern. In this case, the pattern takes over as the maintainer of the Software Instance. If a node with the specified key does not exist, a new Software Instance node is created. In both cases, the Software Instance node is linked to the pattern with a maintainer relationship.

If a key for the Software Instance node is not specified by the pattern, the system creates or updates a group Software Instance with an automatically generated key. The key is based upon the key of the Host upon which the Software Instance is running, the specified type of the Software Instance and, optionally, a key group that can be used to separate the nodes into a number of groups. The count attribute is set to the number of instances in the group identified in the collection of Directly Discovered Data. Each time the host is scanned, the count attribute is changed to represent the number of instances seen in that scan.

**Removal**

Software Instance node can be destroyed either manually or automatically.
To remove a Software Instance node manually, find the necessary Software Instance, select it in the list and pick **Destroy** from the **Actions** list.

Automatic removal happens according to the following scenario. The **age_count** attribute of the (first order) Software Instance node contains information about when the Software Instance node was last confirmed by its maintaining pattern. If the age_count is positive, it represents the number of consecutive scans of the Host node in which the Software Instance was confirmed. If the age_count is negative, it represents the number of consecutive scans in which the Software Instance node was not confirmed. The **last_update_success** and **last_update_failure** attributes contain the date and time at which the Software Instance node was last confirmed, and not confirmed, respectively.

If the pattern does not have a **removal block**, Software Instance nodes are removed using an aging strategy based on the **age_count** and **last_update_success** attributes. The default aging parameters are the same as for a **Host node**, that is, if a Software Instance node has not been seen for at least 7 scans, over a period of at least 10 days, it is destroyed. The parameters can be changed in the options, see Configuring model maintenance settings for more information.

The default aging strategy only applies to Software Instance nodes created from patterns triggering on the following node kinds and maintaining the Software Instances:

- DiscoveredProcess
- DiscoveredService
- DiscoveredListeningPort
- DiscoveredSoftware
- DiscoveredVirtualMachine

If the Software Instance is triggered on anything else, for example, a discovered file, then aging must be implemented in the pattern using a **removal block**.

If the pattern maintaining a node does have a **removal block**, the block can override the default aging scheme to destroy its nodes either earlier or later than normal. For **TKU patterns**, refer to the documentation accompanying each pattern for details of special removal behavior.

Regardless of the presence or absence of a removal block in the pattern, if the Host corresponding to a DDD-triggered Software Instance node is destroyed, the Software Instance node is immediately destroyed (see How nodes get removed).

**Software Instance trigger**

**Creation/update**

When patterns trigger on the creation or modification of other Software Instance nodes, the behavior is simpler. In this situation, the pattern must provide a key for each Software Instance node. The key is used to find an existing Software Instance node to update, or to create a new one. In both cases, the node is linked to the pattern with a maintainer relationship.
Removal

Software Instance nodes created as a result of Software Instance triggers are destroyed using the Cascade removal type; when the triggering Software Instance node is destroyed, the destruction is cascaded to the higher-level Software Instance node. See Cascade Removal.

⚠️ It is possible for other triggers to be used. If any other trigger is used, BMC Atrium Discovery has no automatic removal behavior. Patterns must be used to explicitly destroy any Software Instance nodes created as a result of other triggers.

Software Instance node attributes

The attributes and relationships of a Software Instance node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name name string</td>
<td>Name of the Software Instance.</td>
</tr>
<tr>
<td>Type type string</td>
<td>Type of the Software Instance.</td>
</tr>
<tr>
<td>Instance Count count int</td>
<td>Number of instances grouped together.</td>
</tr>
<tr>
<td>Not displayed in UI key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Instance instance string</td>
<td>The product's own name for this instance.</td>
</tr>
<tr>
<td>VM Type vm_type string</td>
<td>The type of VM technology used by this instance of a virtual machine.</td>
</tr>
<tr>
<td>Cluster Identifier cluster_id string</td>
<td>The internal identifier of the cluster this element is contributing to.</td>
</tr>
<tr>
<td>Cluster Internal Name cluster_name string</td>
<td>The internal name of the cluster this element is contributing to.</td>
</tr>
<tr>
<td>Publisher publisher string</td>
<td>The publisher of the Software Instance. Only populated in cases that a Pattern identifies products whose publisher has changed between released versions, or where more than one publisher may be releasing the software, for example, Apache. This information is normally found in the Pattern node's publishers attribute.</td>
</tr>
<tr>
<td>Product Name product string</td>
<td>The product name. Only populated in cases that a Pattern identifies more than one product. This information is normally found in the Pattern node's products attribute.</td>
</tr>
<tr>
<td>Full Version version string</td>
<td>Full-resolution version.</td>
</tr>
<tr>
<td>Product Version product_version string</td>
<td>Version publicised by the vendor.</td>
</tr>
</tbody>
</table>
### Software Instance node relationships

The relationships on a Software Instance node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>SoftwareInstance: RunningSoftware: HostedSoftware: Host: Host</td>
<td>Host on which this Software Instance is running.</td>
</tr>
<tr>
<td>Host(s)</td>
<td>SoftwareInstance: AggregateSoftware: HostedSoftware: Host: Host</td>
<td>Host(s) on which this aggregate Software Instance is running.</td>
</tr>
<tr>
<td>MFPart</td>
<td>SoftwareInstance: RunningSoftware: HostedSoftware: Host: MFPart</td>
<td>MFPart on which this Software Instance is running.</td>
</tr>
<tr>
<td>MFPart(s)</td>
<td>SoftwareInstance: AggregateSoftware: HostedSoftware: Host: MFPart</td>
<td>MFPart(s) on which this aggregate Software Instance is running.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>SoftwareInstance: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this Software Instance.</td>
</tr>
<tr>
<td>Client to Server Comms</td>
<td>SoftwareInstance: Client: Communication: Server: SoftwareInstance</td>
<td>Server Software Instances that this Software Instance is communicating with.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Server to Client</td>
<td>SoftwareInstance: Server: Communication: Client: SoftwareInstance</td>
<td>Client Software Instances that are communicating with this Software Instance.</td>
</tr>
<tr>
<td>Comms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peer to Peer</td>
<td>SoftwareInstance: Peer: Communication: Peer: SoftwareInstance</td>
<td>Peer Software Instances that are communicating with this Software Instance.</td>
</tr>
<tr>
<td>Comms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contains</td>
<td>SoftwareInstance: SoftwareContainer: SoftwareContainment: ContainedSoftware:</td>
<td>Software instances that make up this aggregate Software Instance.</td>
</tr>
<tr>
<td></td>
<td>SoftwareInstance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SoftwareComponent</td>
<td></td>
</tr>
<tr>
<td>Contained within</td>
<td>SoftwareInstance: ContainedSoftware: SoftwareContainer: SoftwareInstance</td>
<td>Aggregate Software Instances containing this Software Instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Containing Applications</td>
<td>SoftwareInstance: ContainedSoftware: SoftwareContainer: BusinessApplicationInstance</td>
<td>Business applications containing this Software Instance.</td>
</tr>
<tr>
<td>Depends On Cluster</td>
<td>SoftwareInstance: Dependant: Dependency: DependedUpon: Cluster</td>
<td>Cluster this Software Instance is dependant on.</td>
</tr>
<tr>
<td>Database Details</td>
<td>SoftwareInstance: ElementWithDetail: Detail: DatabaseDetail</td>
<td>Details of this Software Instance.</td>
</tr>
<tr>
<td>Software Support Details</td>
<td>SoftwareInstance: ElementWithDetail: SupportDetail: SoftwareDetail</td>
<td>Software Support Detail related to this Software Instance.</td>
</tr>
<tr>
<td>Depends On By Software Instances</td>
<td>SoftwareInstance: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instances that depend upon this Software Instance.</td>
</tr>
<tr>
<td>Depends On Software Instance</td>
<td>SoftwareInstance: DependedUpon: Dependency: DependedUpon: SoftwareInstance</td>
<td>Software Instances that this Software Instance depends on.</td>
</tr>
<tr>
<td>Depends On By Software Components</td>
<td>SoftwareInstance: DependedUpon: Dependency: Dependant: SoftwareComponent</td>
<td>Software Components that depend upon this Software Instance.</td>
</tr>
<tr>
<td>Contained Virtual Host</td>
<td>SoftwareInstance: HostContainer: HostContainment: ContainedHost: Host</td>
<td>Virtual Host provided by this Software Instance.</td>
</tr>
<tr>
<td>Contained MFPart</td>
<td>SoftwareInstance: HostContainer: HostContainment: ContainedHost: MFP</td>
<td>MFPart provided by this Software Instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Detail depending upon this Software Instance.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Depended Upon By</td>
<td>SoftwareInstance: DependedUpon: Dependency: Dependant: Detail</td>
<td>Detail depended upon by this Software Instance.</td>
</tr>
<tr>
<td>Depends On</td>
<td>SoftwareInstance: Dependant: Dependency: DependedUpon: Detail</td>
<td>Detail depended upon by this Software Instance.</td>
</tr>
<tr>
<td>Database Elements This</td>
<td>SoftwareInstance: Dependant: Dependency: DependedUpon: DatabaseDetail</td>
<td>Detail depended upon by this Software Instance.</td>
</tr>
<tr>
<td>Depends On</td>
<td>SoftwareInstance: ElementWithDetail: Detail: Detail</td>
<td>Details of this Software Instance.</td>
</tr>
<tr>
<td>Details</td>
<td>SoftwareInstance: Member: Collection: Collection</td>
<td>A collection which this Software Instance is a member of.</td>
</tr>
<tr>
<td>Managed by Software Instance</td>
<td>SoftwareInstance: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software Instances that manage this Software Instance.</td>
</tr>
<tr>
<td>Manages Software Instances</td>
<td>SoftwareInstance: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software Instances that this Software Instance manages.</td>
</tr>
<tr>
<td>Manages Hosts</td>
<td>SoftwareInstance: ManagedElement: Management: ManagedElement: Host</td>
<td>Hosts that this Software Instance manages.</td>
</tr>
<tr>
<td>Manages Host Containers</td>
<td>SoftwareInstance: ManagedElement: Management: ManagedElement: HostContainer</td>
<td>Host Containers that this Software Instance manages.</td>
</tr>
<tr>
<td>Manages Clusters</td>
<td>SoftwareInstance: ManagedElement: Management: ManagedElement: Cluster</td>
<td>Cluster this Software Instance manages.</td>
</tr>
<tr>
<td>Manages Storage Systems</td>
<td>SoftwareInstance: ManagedElement: Management: ManagedElement: StorageSystem</td>
<td>Storage Systems that this Software Instance manages.</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>SoftwareInstance: ServiceProvider: SoftwareService: Service: LoadBalancerMember</td>
<td>Load balancer member that this Software Instance is implementing a service for.</td>
</tr>
<tr>
<td>Primary processes</td>
<td>SoftwareInstance: InferredElement: Inference: Primary: DiscoveredProcess</td>
<td>Discovered process from which the existence of this Software Instance was inferred.</td>
</tr>
<tr>
<td>Contributor processes</td>
<td>SoftwareInstance: InferredElement: Inference: Contributor: DiscoveredProcess</td>
<td>Discovered process from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Associated processes</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredProcess</td>
<td>Discovered process related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Associated services</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredService</td>
<td>Discovered service related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Contributor: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Contributor: DiscoveredNetworkConnection</td>
<td>Discovered network connection from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Associated network</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered network connection related in some way to this Software Instance.</td>
</tr>
<tr>
<td>connections</td>
<td>Inference: Associate: DiscoveredNetworkConnection</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered listening port from which the existence of this Software Instance was inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Primary: DiscoveredListeningPort</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered listening port from which one or more attributes of this Software Instance were</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: DiscoveredListeningPort</td>
<td>inferred.</td>
</tr>
<tr>
<td>Associated listening</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered listening port related in some way to this Software Instance.</td>
</tr>
<tr>
<td>ports</td>
<td>Inference: Associate: DiscoveredListeningPort</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Package from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: Package</td>
<td></td>
</tr>
<tr>
<td>Associated packages</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Package related in some way to this Software Instance.</td>
</tr>
<tr>
<td></td>
<td>Inference: Associate: Package</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Host from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: Host</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Host related in some way to this Software Instance.</td>
</tr>
<tr>
<td></td>
<td>Inference: Associate: Host</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered file from which the existence of this Software Instance was inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Primary: DiscoveredFile</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered file from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: DiscoveredFile</td>
<td></td>
</tr>
<tr>
<td>Associated files</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered file related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Primary software</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered software from which the existence of this Software Instance was inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Primary: DiscoveredSoftware</td>
<td></td>
</tr>
<tr>
<td>Contributor software</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered software from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: DiscoveredSoftware</td>
<td></td>
</tr>
<tr>
<td>Associated discovered</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered software related in some way to this Software Instance.</td>
</tr>
<tr>
<td>software</td>
<td>Inference: Associate: DiscoveredSoftware</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered command result from which one or more attributes of this Software Instance were</td>
</tr>
<tr>
<td></td>
<td>Inference: Contributor: DiscoveredCommandResult</td>
<td>inferred.</td>
</tr>
<tr>
<td>Associated command</td>
<td>SoftwareInstance: InferredElement:</td>
<td>Discovered command result related in some way to this Software Instance.</td>
</tr>
<tr>
<td>results</td>
<td>Inference: Associate: DiscoveredCommandResult</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Associated registry values</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Associated WMI values</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWMI</td>
<td>Discovered WMI query result related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWMI</td>
<td>Discovered WMI query result from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Associated WBEM instances</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Associated WBEM query results</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM query result related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM query result from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Associated associators results</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association from which one or more attributes of this Software Instance were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Primary: SoftwareInstance</td>
<td>Application whose existence was inferred from this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: Contributor: Inference: Primary: SoftwareInstance</td>
<td>Application whose attributes have been partly or wholly determined from this Software Instance.</td>
</tr>
<tr>
<td>Associated Applications</td>
<td>SoftwareInstance: Associate: Inference: Primary: SoftwareInstance</td>
<td>Application related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: Primary: Inference: Primary: SoftwareInstance</td>
<td>Software whose existence was inferred from this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: Contributor: Inference: Primary: SoftwareInstance</td>
<td>Software whose attributes have been partly or wholly determined from this Software Instance.</td>
</tr>
<tr>
<td>Associated to Software</td>
<td>SoftwareInstance: Associate: Inference: Primary: SoftwareInstance</td>
<td>Software related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Primary: SoftwareInstance</td>
<td>Software whose existence was inferred from this Software Instance.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Contributor: SoftwareInstance</td>
<td>Software whose attributes have been partly or wholly determined from this Software Instance.</td>
</tr>
<tr>
<td>Associated</td>
<td>SoftwareInstance: InferredElement: Inference: Associate: SoftwareInstance</td>
<td>Software related in some way to this Software Instance.</td>
</tr>
<tr>
<td>Software</td>
<td>Not displayed in UI</td>
<td>Inference: Contributor: MFPart</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Primary: MFPart</td>
<td>MFPart from which the existence of this Software Instance was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareInstance: InferredElement: Inference: Contributor: MFPart</td>
<td>MFPart from which one or more attributes of this Software Instance were inferred.</td>
</tr>
</tbody>
</table>

**Software Component node**

A Software Component node represents an instance of a Software Component running inside a Software Instance, for example a deployed application inside a J2EE application server.

**Software Component node lifecycle**

The following section describes the scenarios in which a Software Component node is created, updated or removed.

**Creation/update**

This is under the full control of patterns and as a result there is no default Software Component node behavior.

The key for a Software Component node is entirely dependent on the pattern that creates the Software Component node. It is advised therefore that you take extra care when constructing the key attribute, as it will need to be unique amongst all Software Component nodes. Achieving this uniqueness would typically be done by including the following information in the key:

- The `type` attribute
- The parent node's `key` attribute

**Removal**

Software Component node can be destroyed either manually or automatically.

To remove a Software Component node manually, find the necessary Software Component, select it in the list and pick **Destroy** from the **Actions** list.

Automatic removal happens according to the following scenarios.

- Authoritative removal by the pattern that creates/updates the Software Component node should be considered. The pattern not only needs to create the correct Software Component structure, it also needs to maintain it as the configuration changes
- Built in removal rules will remove all the contained Software Component nodes if an SI/BAI/Host is removed.
Software Component node attributes

The attributes and relationships on a Software Component node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name that the Software Component is known by.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of the Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the Software Component. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Instance</td>
<td>instance string</td>
<td>The product's own name for this instance.</td>
</tr>
<tr>
<td>Full Version</td>
<td>version string</td>
<td>Full-resolution version.</td>
</tr>
<tr>
<td>Product Version</td>
<td>product_version string</td>
<td>Version publicised by the vendor.</td>
</tr>
<tr>
<td>Release</td>
<td>release string</td>
<td>Release number.</td>
</tr>
<tr>
<td>Service Pack</td>
<td>service_pack string</td>
<td>Service pack.</td>
</tr>
<tr>
<td>Build</td>
<td>build string</td>
<td>Build number.</td>
</tr>
<tr>
<td>Patch</td>
<td>patch string</td>
<td>Patch level.</td>
</tr>
<tr>
<td>Revision</td>
<td>revision string</td>
<td>Revision.</td>
</tr>
<tr>
<td>Running</td>
<td>running boolean</td>
<td>Is the software running.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>url string</td>
<td>URL for information about the Software Component. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>business_continuity_critical boolean</td>
<td>If true, the Software Component is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>third_party_support boolean</td>
<td>If true, the Software Component is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>synonyms string</td>
<td>Other names by which this Software Component is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
</tbody>
</table>
## Software Component node relationships

The relationships on a Software Component node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Instance</td>
<td>SoftwareComponent: ContainedSoftware: SoftwareContainment: SoftwareContainer: SoftwareInstance</td>
<td>Software Instance this Software Component is running in.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>SoftwareComponent: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this Software Component.</td>
</tr>
<tr>
<td>Depended Upon By Software Instances</td>
<td>SoftwareComponent: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instances that depend upon this Software Component.</td>
</tr>
<tr>
<td>Depends On Software Instance</td>
<td>SoftwareComponent: Dependant: Dependency: DependedUpon: SoftwareInstance</td>
<td>Software Instances that this Software Component depends on.</td>
</tr>
<tr>
<td>Depended Upon By Software Components</td>
<td>SoftwareComponent: DependedUpon: Dependency: Dependant: SoftwareComponent</td>
<td>Software Components that depend upon this Software Component.</td>
</tr>
<tr>
<td>Depended Upon By Detail</td>
<td>SoftwareComponent: DependedUpon: Dependency: Dependant: Detail</td>
<td>Details that depend upon this Software Component.</td>
</tr>
<tr>
<td>Depends On Detail</td>
<td>SoftwareComponent: Dependant: Dependency: DependedUpon: Detail</td>
<td>Details that this Software Component depends on.</td>
</tr>
<tr>
<td>Load Balancer</td>
<td>SoftwareComponent: ServiceProvider: SoftwareService: Service: LoadBalancerMember</td>
<td>Load balancer member that this Software Component is implementing a service for.</td>
</tr>
<tr>
<td>Primary processes</td>
<td>SoftwareComponent: InferredElement: Inference: Primary: DiscoveredProcess</td>
<td>Discovered process from which the existence of this Software Component was inferred.</td>
</tr>
<tr>
<td>Contributor processes</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: DiscoveredProcess</td>
<td>Discovered process from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated processes</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredProcess</td>
<td>Discovered process related in some way to this Software Component.</td>
</tr>
<tr>
<td>Associated services</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredService</td>
<td>Discovered service related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Discovered network connection</td>
<td>SoftwareComponent: InferredElement: DiscoveredNetworkConnection</td>
<td>Discovered network connection from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated network connections</td>
<td>SoftwareComponent: InferredElement: DiscoveredNetworkConnection</td>
<td>Discovered network connection related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredListeningPort</td>
<td>Discovered listening port from which the existence of this Software Component was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredListeningPort</td>
<td>Discovered listening port from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated listening ports</td>
<td>SoftwareComponent: InferredElement: DiscoveredListeningPort</td>
<td>Discovered listening port related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Package</td>
<td>Package from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated packages</td>
<td>SoftwareComponent: InferredElement: Package</td>
<td>Package related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Host</td>
<td>Host from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Host</td>
<td>Host related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredFile</td>
<td>Discovered file from which the existence of this Software Component was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredFile</td>
<td>Discovered file from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated files</td>
<td>SoftwareComponent: InferredElement: DiscoveredFile</td>
<td>Discovered file related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredFile</td>
<td>Discovered command result from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated command results</td>
<td>SoftwareComponent: InferredElement: DiscoveredCommandResult</td>
<td>Discovered command result related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated registry values</td>
<td>SoftwareComponent: InferredElement: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value related in some way to this Software Component.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: DiscoveredWMI</td>
<td>Discovered WMI query result from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated WMI values</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredWMI</td>
<td>Discovered WMI query result related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated WBEM instances</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM result from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated WBEM query results</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM result related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Associated WBEM associators results</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Contributor: Pattern</td>
<td>Pattern from which one or more attributes of this Software Component were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: Primary: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application whose existence was inferred from this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: Contributor: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application whose attributes have been partly or wholly determined from this Software Component.</td>
</tr>
<tr>
<td>Associated Applications</td>
<td>SoftwareComponent: Associate: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: Primary: Inference: InferredElement: SoftwareInstance</td>
<td>Software whose existence was inferred from this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: Contributor: Inference: InferredElement: SoftwareInstance</td>
<td>Software whose attributes have been partly or wholly determined from this Software Component.</td>
</tr>
<tr>
<td>Associated to Software</td>
<td>SoftwareComponent: Associate: Inference: InferredElement: SoftwareInstance</td>
<td>Software related in some way to this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Primary: SoftwareInstance</td>
<td>Software whose existence was inferred from this Software Component.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SoftwareComponent: InferredElement: Inference: Primary: SoftwareInstance</td>
<td>Software whose existence was inferred from this Software Component.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Associated</td>
<td>SoftwareComponent: InferredElement: Inference: Associate: SoftwareInstance</td>
<td>Software related in some way to this Software Component.</td>
</tr>
<tr>
<td>Software</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Business Application Instance node**

A Business Application Instance (BAI) node is a datastore node which represents a running instance of a known business application running in the environment. This node type is composed of a number of Software Instances and/or other Business Application Instances.

**Business Application Instance lifecycle**

Business Application Instance nodes are created, maintained and destroyed by patterns. Business Application Instance nodes always have a relationship to the Pattern node corresponding to their maintaining pattern.

If a pattern is deleted, any Business Application Instance nodes it is maintaining are immediately destroyed (as are all other nodes it might be maintaining).

**Creation/update**

When a pattern declares the existence of a Business Application Instance node, it must provide a key for it. If the key matches the key of an existing BAI node, the node is updated, otherwise a new BAI node is created.

**Removal**

Business Application Instance node can be destroyed either manually or automatically.

To remove a Business Application Instance node manually, find the necessary Business Application Instance, select it in the list and pick **Destroy** from the **Actions** list.

Automatic removal happens according to the following scenario. Business Application Instance node patterns normally trigger on the creation or modification of Software Instance or Business Application Instance nodes. In this case, the Business Application Instance nodes are removed using the Cascade removal type; when the triggering Software Instance node or Business Application Instance node is destroyed, the destruction is cascaded to the Business Application Instance. See **Cascade Removal**.

If a Business Application Instance node is created by a pattern triggered on a node kind other than a Software Instance node or Business Application Instance node, BMC Atrium Discovery has no automatic removal behavior. Patterns must be used to explicitly destroy any such Business Application Instance node.
**Business Application Instance attributes**

The attributes and relationships on a Business Application Instance node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name that the Business Application is known by.</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Business Application type.</td>
</tr>
<tr>
<td>Instance</td>
<td>instance string</td>
<td>The application's own name for this instance.</td>
</tr>
<tr>
<td>Full Version</td>
<td>version string</td>
<td>Full-resolution version.</td>
</tr>
<tr>
<td>Product Version</td>
<td>product_version string</td>
<td>Version publicised by the vendor.</td>
</tr>
<tr>
<td>Release</td>
<td>release string</td>
<td>Release name.</td>
</tr>
<tr>
<td>Service Pack</td>
<td>service_pack string</td>
<td>Service pack.</td>
</tr>
<tr>
<td>Build</td>
<td>build string</td>
<td>Build number.</td>
</tr>
<tr>
<td>Patch</td>
<td>patch string</td>
<td>Patch level.</td>
</tr>
<tr>
<td>Revision</td>
<td>revision string</td>
<td>Revision.</td>
</tr>
<tr>
<td>Support Email</td>
<td>support_email string</td>
<td>Email contact details for the support of this Business Application Instance.</td>
</tr>
<tr>
<td>Support Phone</td>
<td>support_phone string</td>
<td>Telephone contact details for the support of this Business Application Instance.</td>
</tr>
<tr>
<td>Support Website</td>
<td>support_website string</td>
<td>Website details for the support of this Business Application Instance.</td>
</tr>
<tr>
<td>Not shown in the UI</td>
<td>_explicit_removal _string</td>
<td>Attributes used internally for explicit removal.</td>
</tr>
</tbody>
</table>

**Not shown in the UI**

| _explicit_removal _string | Attributes used internally for explicit removal. |
### UI Name

<table>
<thead>
<tr>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description of the Business Application. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>URL</td>
<td>URL for information about the Business Application. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>If true, the Business Application is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>If true, the Business Application is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>Synonyms</td>
<td>Other names by which this Business Application is known. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
</tbody>
</table>

### Business Application Instance relationships

The relationships on a Business Application Instance node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applications This Depends On</td>
<td>BusinessApplicationInstance: Dependant: Dependency: DependedUpon: BusinessApplicationInstance</td>
<td>Applications on which this application depends.</td>
</tr>
<tr>
<td>Applications Depending Upon This</td>
<td>BusinessApplicationInstance: DependedUpon: Dependency: Dependant: BusinessApplicationInstance</td>
<td>Applications that depend on this application.</td>
</tr>
<tr>
<td>Database Elements This Depends On</td>
<td>BusinessApplicationInstance: DependedUpon: Dependency: DependedUpon: DatabaseDetail</td>
<td>Detail that this application depends on.</td>
</tr>
<tr>
<td>Location</td>
<td>BusinessApplicationInstance: ElementInLocation: Location: Location: Location</td>
<td>Location of this application.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: ElementInCategory: ElementCategory: Category: Family</td>
<td>Family of this application.</td>
</tr>
<tr>
<td>Hosts</td>
<td>Hosts where this instance is running.</td>
<td></td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MFParts</td>
<td>BusinessApplicationInstance: AggregateSoftware: HostedSoftware: Host: MFPart</td>
<td>MFParts where this instance is running.</td>
</tr>
<tr>
<td>Details</td>
<td>BusinessApplicationInstance: ElementWithDetail: Detail: Detail: Detail</td>
<td>Details of this application.</td>
</tr>
<tr>
<td>Collections</td>
<td>BusinessApplicationInstance: Member: Collection: Collection</td>
<td>A collection which this Business Application Instance is a member of.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this application instance.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Primary: SoftwareInstance</td>
<td>Software instance from which the existence of this application was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Contributor: SoftwareInstance</td>
<td>Software instance from which one or more attributes of this application were inferred.</td>
</tr>
<tr>
<td>Associated Software</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Associate: SoftwareInstance</td>
<td>Software instance related in some way to this application.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Primary: BusinessApplicationInstance</td>
<td>Application instance from which the existence of this application was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Contributor: BusinessApplicationInstance</td>
<td>Application instance from which one or more attributes of this application were inferred.</td>
</tr>
<tr>
<td>Associated Applications</td>
<td>BusinessApplicationInstance: InferredElement: Inference: Associate: BusinessApplicationInstance</td>
<td>Application instance related in some way to this application.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: Primary: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application instance related in some way to this application.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>BusinessApplicationInstance: Contributor: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application instance related in some way to this application.</td>
</tr>
<tr>
<td>Associated to Applications</td>
<td>BusinessApplicationInstance: Associate: Inference: InferredElement: BusinessApplicationInstance</td>
<td>Application instance related in some way to this application.</td>
</tr>
</tbody>
</table>
Functional Component node

A Functional Component node represents an application component created using Collaborative Application Mapping. They represent a grouping of software "items", or parts of the application, into a logical collection that provides an element of the Business Application Instance.

Examples in a tiered application are, Web tier, or Database tier.

Functional Component node lifecycle

The following section describes the scenarios in which a Functional Component node is created, updated or removed.

Creation/update

Functional Component nodes are created and updated in exactly the same way, and in parallel with their related BAI nodes.

Removal

Functional Component nodes are removed in exactly the same way, and in parallel with their related BAI nodes.

Functional Component Attributes

The attributes on a Functional Component are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name of the functional component.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of functional component.</td>
</tr>
<tr>
<td>Required</td>
<td>is_required boolean</td>
<td>True if the functional component is required.</td>
</tr>
<tr>
<td>Instance</td>
<td>instance string</td>
<td>The component's own name for this instance.</td>
</tr>
</tbody>
</table>

Functional Component Relationships

The relationships on a Functional Component node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detail</td>
<td>FunctionalComponent: FunctionalContainer: FunctionalContainment: ContainedFunctionality: Detail</td>
<td>Detail that is a member of this Functional Component</td>
</tr>
<tr>
<td>Database Detail</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Mainframe node

A Mainframe node represents a mainframe computer Central Electronic Complex (CEC) or Central Processing Complex (CPC).

#### Mainframe node lifecycle

The following section describes the scenarios in which a Mainframe node is created, updated, or removed.

**Creation/update**

Mainframe nodes are created and updated in exactly the same way as Host nodes.

**Removal**

Mainframe nodes are removed in exactly the same way as Host nodes.

#### Mainframe node attributes

The attributes and relationships on a Mainframe node are described in the tables below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>FunctionalComponent: FunctionalContainer:</td>
<td>The name that the Mainframe is known by.</td>
</tr>
<tr>
<td></td>
<td>FunctionalContainment: ContainedFunctionality:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DatabaseDetail</td>
<td>Database Detail that is a member of this Functional Component.</td>
</tr>
<tr>
<td>Software</td>
<td>FunctionalComponent: FunctionalContainer:</td>
<td>Software Component that is a member of this Functional Component.</td>
</tr>
<tr>
<td>Component</td>
<td>FunctionalContainment: ContainedFunctionality:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SoftwareComponent</td>
<td></td>
</tr>
<tr>
<td>Software</td>
<td>FunctionalComponent: FunctionalContainer:</td>
<td>Software Instance that is a member of this Functional Component.</td>
</tr>
<tr>
<td>Instance</td>
<td>FunctionalContainment: ContainedFunctionality:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SoftwareInstance</td>
<td></td>
</tr>
<tr>
<td>Business</td>
<td>FunctionalComponent: FunctionalContainer:</td>
<td>Business Application Instance that is a member of this Functional Component.</td>
</tr>
<tr>
<td>Application</td>
<td>FunctionalContainment: ContainedFunctionality:</td>
<td></td>
</tr>
<tr>
<td>Instance</td>
<td>BusinessApplicationInstance</td>
<td></td>
</tr>
<tr>
<td>Containing</td>
<td>FunctionalComponent: ContainedFunctionality:</td>
<td>Business Application Instance containing this Functional Component.</td>
</tr>
<tr>
<td>Business</td>
<td>FunctionalContainer: ContainedFunctionality:</td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>BusinessApplicationInstance</td>
<td></td>
</tr>
<tr>
<td>Maintaining</td>
<td>FunctionalComponent: Element: Maintainer: Pattern:</td>
<td>Pattern that is maintaining this Functional Component.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Pattern</td>
<td></td>
</tr>
</tbody>
</table>

Mainframe node attributes

### Mainframe node attributes

- **Name**
  - **Type**: string
  - **Description**: The name that the Mainframe is known by.

- **Description**
  - **Type**: string
  - **Description**: Description of the Mainframe. Legacy attribute not currently used. Can be used by patterns if desired.
### UI Name and Attribute Name and Type

<table>
<thead>
<tr>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>model string</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
</tr>
<tr>
<td>Serial</td>
<td>serial string</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>num_processors string</td>
</tr>
<tr>
<td>Available CPs</td>
<td>available_cps int</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms string</td>
</tr>
</tbody>
</table>

### Mainframe node relationships

The relationships on a Mainframe node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Members</td>
<td>Mainframe: HostContainer: HostContainment: ContainedHost: MFPart</td>
<td>MFParts that are within this Mainframe.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Mainframe: InferredElement: Inference: Primary: DiscoveredMainframe</td>
<td>DiscoveredMainframe for this Mainframe.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Mainframe: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this Network Device.</td>
</tr>
<tr>
<td>Destination Used To Be Identified As</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Mainframe: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>Mainframe: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>Mainframe: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Family</td>
<td>Mainframe: ElementInCategory: ElementCategory: Category</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>Mainframe: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>Mainframe: ElementInLocation: Location: Location: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Support Manager</td>
<td>Mainframe: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>Mainframe: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>Mainframe: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Mainframe: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
</tbody>
</table>

**MFPart node**

An MFPart node represents a mainframe OSI either as a native LPAR running zOS, or a zVM-supported VM running zOS.

**MFPart node lifecycle**

The following section describes the scenarios in which an MFPart node is created, updated, or removed.

**Creation and update**

MFPart nodes are created and updated in exactly the same way as Host nodes.

**Removal**

MFPart nodes are removed in exactly the same way as Host nodes.

An MFPart node is also destroyed when its parent mainframe is destroyed. This is a Containment Removal type because an MFPart cannot exist without its mainframe host. See Containment Removal. For information about the destruction of Host nodes, see Host node lifecycle.

**MFPart node attributes**

The attributes and relationships on a MFPart node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of mainframe part.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of mainframe part.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Model of mainframe part.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Vendor of mainframe part.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>OS</td>
<td>os string</td>
<td>Operating system.</td>
</tr>
<tr>
<td>OS Type</td>
<td>os_type string</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>OS Class</td>
<td>os_class string</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version string</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>OS System Name</td>
<td>os_serial string</td>
<td>Operating system name.</td>
</tr>
<tr>
<td>Virtual</td>
<td>virtual string</td>
<td>Flag if the MFPart is virtual.</td>
</tr>
<tr>
<td>Partition</td>
<td>partition string</td>
<td>Flag if the MFPart is a partition.</td>
</tr>
<tr>
<td>VM ID</td>
<td>vm_id string</td>
<td>The virtual machine ID.</td>
</tr>
<tr>
<td>Part Name</td>
<td>mfpart_name string</td>
<td>Mainframe part name.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>part_type string</td>
<td>Mainframe part type. This attribute is deprecated, use type instead.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Mainframe part type.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>age_count int</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_success date</td>
<td>The time at which a scan was last successfully associated with this MFPart.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_failure date</td>
<td>The time at which a scan associated with this MFPart failed.</td>
</tr>
<tr>
<td>Last successful CMDB sync</td>
<td>last_cmdb_sync_success date</td>
<td>The time at which this MFPart was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>last_cmdb_sync_failure date</td>
<td>The time at which an attempt to synchronize this MFPart into the CMDB failed.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>last_cmdb_sync_duration float</td>
<td>The time in seconds spent performing the last CMDB synchronization of this MFPart.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>last_cmdb_sync_ci_count int</td>
<td>The number of CIs corresponding to this MFPart at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>last_cmdb_sync_rel_count int</td>
<td>The number of relationships between CIs corresponding to this MFPart at the last CMDB synchronization.</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
</tbody>
</table>
## UI Name and Type

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name string</td>
<td>The primary name.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Short Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>short_name string</td>
<td>The short name.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>description string</td>
<td>The description of the element.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url string</td>
<td>The URL for information about the element.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Continuity Critical</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>business_continuity_critical : boolean</td>
<td>If true, element is critical to operation of the business.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Supported by 3rd Party</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>third_party_support : boolean</td>
<td>True if the element is supported by a third party.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Synonyms</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>synonyms : list string</td>
<td>Other names for the element.</td>
</tr>
</tbody>
</table>

## MFPart node relationships

The relationships on a MFPart node are described in the following table.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Instances</td>
<td>MFPart: Host: HostedSoftware: RunningSoftware: SoftwareInstance</td>
<td>Software Instances running on this MFPart.</td>
</tr>
<tr>
<td>Aggregate Software Instances</td>
<td>MFPart: Host: HostedSoftware: AggregateSoftware: SoftwareInstance</td>
<td>Aggregate Software Instances running on this MFPart.</td>
</tr>
<tr>
<td>Hosted Applications</td>
<td>MFPart: Host: HostedSoftware: AggregateSoftware: BusinessApplicationInstance</td>
<td>Business Application Instances running on this MFPart.</td>
</tr>
<tr>
<td>Mainframe</td>
<td>MFPart: ContainedHost: HostContainment: HostContainer: Mainframe</td>
<td>Mainframe of which this is a member.</td>
</tr>
<tr>
<td>Sysplex</td>
<td>MFPart: ContainedHost: HostContainment: HostContainer: Cluster</td>
<td>Sysplex of which this is a member.</td>
</tr>
<tr>
<td>CouplingFacility</td>
<td>MFPart: Host: CouplingFacility: CouplingFacility: CouplingFacility</td>
<td>CouplingFacility of which this is a member.</td>
</tr>
<tr>
<td>Storage Collection</td>
<td>MFPart: Host: Storage: Storage: StorageCollection</td>
<td>Storage Collections used by this MFPart.</td>
</tr>
<tr>
<td>Storage</td>
<td>MFPart: Host: Storage: Storage: StorageCollection</td>
<td>Storage used by this MFPart.</td>
</tr>
<tr>
<td>Containing VM</td>
<td>MFPart: ContainedHost: HostContainment: HostContainer: SoftwareInstance</td>
<td>Software container for this MFPart.</td>
</tr>
<tr>
<td>Discovery Access</td>
<td>MFPart: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this MFPart.</td>
</tr>
</tbody>
</table>
### Coupling Facility node

A Coupling Facility node represents the mainframe computer's coupling facility, the component of the mainframe that enables multiple LPARs to access the same data.

### Coupling Facility node lifecycle

The following section describes the scenarios in which a Coupling Facility node is created, updated, or removed.

#### Creation and update

Coupling Facility nodes are created and updated in exactly the same way as Host nodes.

#### Removal

A Coupling Facility node can be destroyed either manually or automatically.
To remove a Coupling Facility node manually, find the necessary Coupling Facility, select it in the list and pick **Destroy** from the **Actions** list.

Automatic removal happens according to the following scenario.

Coupling Facility nodes are removed when no relationships to **MFPart nodes** remain.

A Coupling Facility node is also destroyed when its parent mainframe is destroyed. This is a Containment Removal type because a CouplingFacility cannot exist without its mainframe host. See Containment Removal. For information about the destruction of Host nodes, see Host node lifecycle.

**CouplingFacility node attributes**

The attributes and relationships on a Coupling Facility node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>The primary name that the Coupling Facility is known by.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>The short name.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the Coupling Facility. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
<td>URL for information about the Coupling Facility. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
<td>If true, the Coupling Facility is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
<td>If true, the Coupling Facility is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms string</td>
<td>Other names by which this Coupling Facility is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Internal Storage</td>
<td>storage string</td>
<td>Internal Storage capacity (bytes).</td>
</tr>
<tr>
<td>Resource Manager Policy</td>
<td>cfrm_name string</td>
<td>Resource Manager Policy.</td>
</tr>
<tr>
<td>Node Descriptor</td>
<td>node_descriptor string</td>
<td>Node descriptor.</td>
</tr>
</tbody>
</table>

**CouplingFacility node relationships**

The relationships on a Coupling Facility node are described in the table below.
## Cluster node

A Cluster node represents a group of hosts collaborating to form a cluster. The Cluster node represents the group rather than a single physical entity. They are also used to represent mainframe sysplexes. Cluster nodes are under the full control of patterns.

For detailed information about patterns, see the [The Pattern Language TPL](#) and [Configipedia](#).

### Cluster node lifecycle

The following section describes the scenarios in which a Cluster node is created, updated or removed.
Creation
The creation of a Cluster node is under the full control of patterns. Clusters are created when they are detected on a host by a pattern which is searching for them. The pattern contains the conditions which enable it to detect a cluster. Once found, the pattern creates the Cluster that it needs. A relationship link is automatically created to the Host node and the Cluster will have a key generated.

The generated key for a Cluster node is entirely dependent on the kind of Cluster.

Update
The update procedure for a Cluster node is also under full control of patterns. When the pattern has identified a cluster, it calculates a key. If a Cluster node with that key already exists, it is updated accordingly.

Removal
A Cluster node can be destroyed either manually or automatically.

To remove a Cluster node manually, find the necessary Cluster, select it in the list and pick Destroy from the Actions list.

Automatic removal happens according to the following scenario.

The default removal of a Cluster node is not under the control of patterns. A Cluster node will be removed when the last remaining Host node or MFPart is unlinked. In some cases, particular patterns might not create a cluster until hosts are detected. In these cases, the pattern will remove the Cluster node when there is only one host linked to it still or in the case of a vCenter cluster node when the evidence for the VMware cluster is not found, the Cluster node is deleted.

This is a Cascade Removal type, see Cascade Removal.

Cluster node attributes
The attributes and relationships on a Cluster node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Details</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of cluster.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>id string</td>
<td>Internal Identifier of cluster.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internal Name</td>
<td>cluster_name string</td>
<td>Internal Name of cluster</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>The primary name that the cluster is known by.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the cluster. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
<td>URL for information about the cluster. Legacy attribute not currently used. Can be used by patterns if desired</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
<td>If true, the cluster is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
<td>If true, the cluster is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms list:string</td>
<td>Other names by which this cluster is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
</tbody>
</table>

**Cluster node relationships**

The relationships on a Cluster node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member Hosts</td>
<td>Cluster: HostContainer: HostContainer: ContainedHost: Host</td>
<td>Logical hosts that are part of this logical cluster.</td>
</tr>
<tr>
<td>Member MFParts</td>
<td>Cluster: HostContainer: HostContainer: ContainedHost: MFPart</td>
<td>MFParts that are part of this Sysplex.</td>
</tr>
<tr>
<td>Member Coupling Facilities</td>
<td>Cluster: HostContainer: HostContainer: CouplingFacility: CouplingFacility</td>
<td>Coupling Facilities that are part of this Sysplex.</td>
</tr>
<tr>
<td>Service Software</td>
<td>Cluster: Service: SoftwareService: ServiceProvider: SoftwareInstance</td>
<td>Software instances that provide this logical cluster.</td>
</tr>
<tr>
<td>Dependant Software</td>
<td>Cluster: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software instances that depend upon this logical cluster.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>Cluster: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this cluster.</td>
</tr>
<tr>
<td>Managing Software</td>
<td>Cluster: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software instances that is managing this cluster.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Cluster: AttachmentContainer: Attachment: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>Cluster: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>Cluster: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
</tbody>
</table>
File node

A File node is most often used to track a configuration file. The files that are tracked for each OS are specified by patterns. File nodes are under the full control of patterns.

For detailed information about patterns, see the The Pattern Language TPL and Configipedia.

File node lifecycle

The following section describes the scenarios in which a File node is created, updated or removed.

Creation/update

This is under the full control of patterns and as a result there is no default File node behavior. The generated key for a File node is entirely dependent on the pattern that creates the File node.

Removal

A File node can be destroyed in one of following ways:

- Explicitly by the pattern that created it. See Removal in the Pattern Language guide
- If the File node is triggered from another Inferred node, then when this inferred node is destroyed, so will the File node. This is a Cascade Removal type, see Cascade Removal.
- Manually, by selecting the necessary File node, and picking Destroy from the Actions list.

File node attributes

The attributes and relationships on a File node are described in the tables below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family</td>
<td>Cluster: ElementInCategory: ElementCategory: Category: Family</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>Cluster: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>Cluster: ElementInLocation: Location: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>Cluster: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>Cluster: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>Cluster: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path path string</td>
<td>The absolute path to the tracked configuration file.</td>
</tr>
</tbody>
</table>
### UI Name and Type

<table>
<thead>
<tr>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size size int</td>
<td>The size (in bytes) of the tracked configuration file.</td>
</tr>
<tr>
<td>MD5 Checksum md5sum string</td>
<td>The md5 checksum of the file.</td>
</tr>
<tr>
<td>Last Modified last_modified string</td>
<td>The time and date of the last modification to tracked configuration file. This date and time is taken from the file’s timestamp on the host machine.</td>
</tr>
<tr>
<td>Content content string</td>
<td>The raw contents of the tracked configuration file. This attribute is populated by a cat command on UNIX or a type command in Windows. Binary files can be modelled, but the contents will not included.</td>
</tr>
</tbody>
</table>

### File node relationships

The relationships on a File node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>File: HostedFile: HostedFile: Host: Host</td>
<td>Hosts on which this File node is tracked.</td>
</tr>
<tr>
<td>Depended Upon By</td>
<td>File: DependedUpon: Dependency: Dependant: Detail</td>
<td>Details depending on this file.</td>
</tr>
</tbody>
</table>

### File System node

A File System node represents a file system mounted on a host. On a host there is a File System node for each discovered local, exported, and remote file system.

Currently File System nodes are not synced to CMDB by the default syncmapping sets. This can be extended.

### File System Model Illustration

![File System Model Illustration](image)

FileSystem nodes come in the following types:
• **LOCAL** — A filesystem that is LOCAL to the device, for instance a directly attached disk drive. SAN storage appears to the OS as LUNs on a SCSI chain filesystem, so these devices are reported as LOCAL.

• **REMOTE** — A filesystem that is attached over the network from another device. This is the client side of a network filesystem.

• **EXPORTED** — A filesystem (or part of a filesystem) which is available for other devices to use. This is the server side of a network filesystem.

An important aspect of this model is that the REMOTE type is the client's view of an EXPORTED filesystem. If both server and client ends are discovered there is a single EXPORTED FileSystem node attached to the server Host node and multiple REMOTE FileSystem instances, a single instance attached to each client Host node.

Network file systems generally export a fraction of an underlying LOCAL filesystem. Thus, on a server exporting filesystems it is expected that the LOCAL and EXPORTED FileSystems instances will be related by the `FileSystem:Local:NetworkFileSystem:Exported:FileSystem` relationship showing which LOCAL filesystem supports the EXPORTED filesystem. There might be several EXPORTED FileSystem nodes related to the same LOCAL FileSystem node, for instance on a Windows server the two shares `C$` and `ADMIN$` will usually both be exported from the same `C:` local filesystem.

The above diagram shows a client (right side) and server (left side) Host. The server might be a UNIX server exporting a section of a local filesystem via NFS, or a Windows server exporting a shared folder via SMB. For each client mounting the NFS/SMB filesystem, the filesystem will be represented by a REMOTE FileSystem.

⚠️ It is important to remember this distinction if you are considering synchronizing these nodes to the CMDB. The BMC Atrium Discovery model and the Common Data Model (CDM) have very different semantics around "remote" filesystems. Currently it is recommended to simply synchronize these to `BMC_FileSystem`.

**File System Lifecycle**

A File System node is related to the host it is contained in. Therefore, the File System node lifecycle is directly tied to the Host node and is destroyed when the Host node is destroyed. See [Host node](#).

This is a Containment Removal type, see [Containment Removal](#).

Additionally if both ends of a network filesystem can be discovered the system builds a `NetworkFileSystem` relationship between the appropriate REMOTE and EXPORTED FileSystem nodes.
File System node attributes

The attributes and relationships of a File System node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type; Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>File system unique identifier.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Meaningful file system name for reporting.</td>
</tr>
<tr>
<td>Filesystem Name</td>
<td>fs_name string</td>
<td>File system name.</td>
</tr>
<tr>
<td>Kind</td>
<td>fs_kind string</td>
<td>File system kind.</td>
</tr>
<tr>
<td>Type</td>
<td>fs_type string</td>
<td>File system type.</td>
</tr>
<tr>
<td>Mount</td>
<td>mount string</td>
<td>File system mount.</td>
</tr>
<tr>
<td>Size</td>
<td>size int</td>
<td>File system size in KB.</td>
</tr>
<tr>
<td>Used</td>
<td>used int</td>
<td>File system used in KB.</td>
</tr>
<tr>
<td>Free</td>
<td>free int</td>
<td>File system free in KB.</td>
</tr>
<tr>
<td>Used Percentage</td>
<td>used_percentage int</td>
<td>File system used percentage.</td>
</tr>
<tr>
<td>Free Percentage</td>
<td>free_percentage int</td>
<td>File system free percentage.</td>
</tr>
<tr>
<td>Remote Hostname</td>
<td>remote_hostname string</td>
<td>Hostname of remote server for REMOTE File Systems.</td>
</tr>
<tr>
<td>Remote IP Address</td>
<td>remote_ip string</td>
<td>IP Address of remote server for REMOTE File Systems.</td>
</tr>
<tr>
<td>Remote Username</td>
<td>remote_username string</td>
<td>Username used to access REMOTE File System.</td>
</tr>
<tr>
<td>Remote Name</td>
<td>remote_fs_name string</td>
<td>Remote file system name.</td>
</tr>
<tr>
<td>Exported Path</td>
<td>path string</td>
<td>Exported path.</td>
</tr>
<tr>
<td>Has Qtrees</td>
<td>has_qtrees boolean</td>
<td>Has related Qtrees.</td>
</tr>
<tr>
<td>Is a Qtree</td>
<td>qtree boolean</td>
<td>Is a Qtree.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type; Target</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Owning Volume</td>
<td>root_volume string</td>
<td>Root Volume.</td>
</tr>
</tbody>
</table>

**File System node relationships**

The relationships of a File System node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inference</td>
<td>InferredElement: Inference: Primary: DiscoveredFileSystem</td>
<td>Primary inference.</td>
</tr>
<tr>
<td>Disk Drive</td>
<td>FileSystem: Consumer: StorageUse: Provider: DiskDrive</td>
<td>DiskDrive providing storage.</td>
</tr>
</tbody>
</table>

**Package node**

A Package node represents a software package in your environment and is linked to all Host nodes which contain it.

**Package node lifecycle**

The following section describes the scenarios in which a Package is created, updated or destroyed. Package nodes are identical in behavior to Patch nodes, see [Patch node](#).

**Creation**

Package nodes are created when they are first detected on a host. The generated key of a Package node consists of the following critical properties:

- name
- version
- revision
- epoch
• architecture (for example: i386, x86_64)
• OS information.

Package nodes are shared by all host nodes with the same package installed, the same architecture and that are running the same OS.

Update
A Package node is never updated because its key consists of the critical properties described above, and a change to any of these properties will result in a new Package node being created.

Removal
To remove a Package node manually, find the necessary Package node, select it in the list and pick **Destroy** from the **Actions** list.

Package node attributes
The attributes and relationships of a Package node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Package vendor.</td>
</tr>
<tr>
<td>Version</td>
<td>version string</td>
<td>Package version.</td>
</tr>
<tr>
<td>Epoch</td>
<td>epoch int</td>
<td>Package epoch.</td>
</tr>
<tr>
<td>Revision</td>
<td>revision string</td>
<td>Package revision.</td>
</tr>
<tr>
<td>Operating System</td>
<td>os string</td>
<td>Operating System this Package is for. Derived from the os_type attribute of the Host nodes it is installed on.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Unique key.</td>
</tr>
<tr>
<td>Architecture</td>
<td>arch string</td>
<td>CPU type Package runs under.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Package description.</td>
</tr>
<tr>
<td>Solaris Full Name</td>
<td>pkname string</td>
<td>Full Solaris package name.</td>
</tr>
</tbody>
</table>

Package node relationships
The relationships of a Package node are described in the table below.
<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hosts</td>
<td>Package: InstalledSoftware; HostedSoftware: Host</td>
<td>Hosts with this package.</td>
</tr>
<tr>
<td></td>
<td>Host: Host</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Package: Associate: Inference:</td>
<td>Software related in some way to this package.</td>
</tr>
<tr>
<td></td>
<td>Inferrelement: SoftwareInstance</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Package: Contributor: Inference:</td>
<td>Software whose attributes have been partly or wholly determined from this package.</td>
</tr>
<tr>
<td></td>
<td>Inferrelement: SoftwareInstance</td>
<td></td>
</tr>
</tbody>
</table>

**Patch node**

A Patch node represents a software patch in your network environment and is linked to any Host node which contains it.

⚠️ You will only find Patch nodes linked to operating systems which recognize patches. Currently only three operating systems that BMC Atrium Discovery models have the concept of a patch; Solaris, HPUX and Windows.

**Patch node lifecycle**

The following section describes the scenarios in which a Patch is created, updated or destroyed. Patch nodes are identical in behavior to Package nodes, see [Package node](#).

**Creation**

Patch nodes are created when they are first detected on a host. The key of a Patch node consists of two critical properties; name and OS information. Therefore, Patch nodes are shared by all Host nodes with the same patch installed.

**Update**

A Patch node is never updated because its key consists of two critical properties and a change to either of them will result in a new Patch node being created.

**Removal**

To remove a Patch node manually, find the necessary Patch node, select it in the list and pick **Destroy** from the **Actions** list.

**Patch node attributes**

The attributes and relationships of a Patch node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Patch name.</td>
<td></td>
</tr>
</tbody>
</table>
### Patch node relationships

The relationships of a Patch node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>

### Fibre Channel nodes

*Fibre Channel nodes Terminology*

The term Fibre Channel nodes is used to refer collectively to the three node kinds that are used to model connections to a SAN.

A Fibre Channel Host Bus Adaptor (HBA) is a controller card that connects a host to a storage area network (SAN). The controller card contains logical devices that process data for transmission. Each logical device has a number of ports which are the hardware attachments that allow the node to send and receive information.

The following nodes are used to model connections to a SAN:

- HBA node — models the Host Bus Adaptor card
- FCN node — models the logical devices
- FCP node — models the ports

This is shown below.
This diagram shows the relationship cascading from one HBA node to two child FCN nodes with two child FCP nodes.

These nodes all have the same lifecycle, that is, they are created, or removed at the same time, as the result of the same discovery data. The same data also updates the nodes, though not all nodes might change at the same time. The lifecycle for all three node types is shown in the following sections, and the attributes and links are shown for individual node types. The three node types are referred to collectively as Fibre Channel nodes.

BMC Atrium Discovery currently models QLogic and Emulex HBA cards.

**HBA node**

An HBA node represents a Fibre Channel Host Bus Adaptor, the controller card that connects a host, usually via an optical connection to a SAN. The HBA node has child nodes; one or more FCN nodes, which in turn have child nodes; one or more FCP nodes.

**FCN node**

The FCN node represents the logical devices that process data for transmission. The logical devices are called FCNs and each has a unique World Wide Node Number (WWNN). FCNs have firmware on board which might be updated frequently. The firmware version can differ between nodes on the same HBA card.

**FCP node**

FCN nodes have one or more ports to provide a physical interface to communicate. The port is a hardware attachment that allows the node to send and receive information via the physical interface. The ports are called FCPs and each has a unique World Wide Port Number (WWPN).

**Fibre Channel nodes lifecycle**

The following section describes the scenarios in which the Fibre Channel nodes are created, updated or destroyed.
Creation
The Fibre Channel nodes are created in the datastore when a host is scanned with sufficient credentials to run the HBA discovery commands. See the GetHBAList methods in User privileges and information access for UNIX and related operating systems and User privileges and information access for Windows operating systems. New HBA nodes and their children are only created if an HBA node with the same name (hba_id) does not exist on that Host node.

Update
The Fibre Channel nodes are updated in the datastore when a host is scanned with sufficient credentials to run the HBA discovery commands.

Removal
The Fibre Channel nodes are not automatically removed, by default. They are only removed if their parent Host node is destroyed. This is a Containment Removal type, see Containment Removal. To remove a Fibre Channel node manually, find the necessary Fibre Channel node, select it in the list and pick Delete from the Actions list.

Fibre Channel HBA attributes
The attributes and relationships on an HBA node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HBA Identifier</td>
<td>hba_id string</td>
<td>The HBA identifier. For example 20:00:00:00:c9:38:dc.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>The HBA vendor. For example, Emulex.</td>
</tr>
<tr>
<td>Firmware</td>
<td>firmware list:string</td>
<td>A list of firmware versions present on FCNs on this HBA. For example TS1.91A0. FCNs on the same HBA can have different firmware versions.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Model name.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Model description.</td>
</tr>
<tr>
<td>Serial</td>
<td>serial string</td>
<td>Serial number.</td>
</tr>
<tr>
<td>Driver Version</td>
<td>driver_version string</td>
<td>Driver version.</td>
</tr>
<tr>
<td>Optrom Bios Version</td>
<td>optrom_bios_version string</td>
<td>Optrom bios version.</td>
</tr>
<tr>
<td>Optrom Efi Version</td>
<td>optrom_efi_version string</td>
<td>Optrom efi version.</td>
</tr>
</tbody>
</table>
UI Name | Attribute Name and Type | Description
--- | --- | ---
Optrom Fcode Version | optrom_fcode_version | Optrom fcode version.
Optrom Fw Version | optrom_fw_version | Optrom fw version.
Option ROM Version | option_rom_version | Optional ROM version.
Driver Name | driver_name | Driver name.
Boot Bios | boot_bios | Boot bios.
AIX Part Number | aix_part_number | AIX part number.
AIX Manufacturer Code | aix_manufacturer_code | AIX manufacturer number.

Fibre Channel HBA relationships

The relationships of an HBA node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>FibreChannelHBA: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: Host</td>
<td>Relationship to the host with this HBA.</td>
</tr>
<tr>
<td>Fibre Channel Node</td>
<td>FibreChannelHBA: FibreChannelDeviceWithNode: FibreChannelNodeDevice: FibreChannelNode: FibreChannelNode</td>
<td>Relationship to one or more FCNs (WWNNs) on this HBA card.</td>
</tr>
<tr>
<td>Discovered HBA</td>
<td>FibreChannelHBA: InferredElement: Inference: Primary: DiscoveredHBA</td>
<td>Primary inference.</td>
</tr>
</tbody>
</table>

FCN node attributes

The attributes and relationships on an FCN node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WWNN</td>
<td>wwnn</td>
<td>World Wide Node Name. The identifier for this FCN.</td>
</tr>
<tr>
<td>Firmware</td>
<td>firmware</td>
<td>The firmware present on this FCN. This can differ between FCNs on the same HBA.</td>
</tr>
</tbody>
</table>
FCN node relationships

The relationships of an FCN node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fibre Channel HBA</td>
<td>FibreChannelNode: FibreChannelNodeDevice: FibreChannelDeviceWithNode: FibreChannelHBA</td>
<td>The HBA for this FCN.</td>
</tr>
<tr>
<td>Fibre Channel Port</td>
<td>FibreChannelNode: FibreChannelNodeWithPort: FibreChannelNodePort: FibreChannelPort</td>
<td>Ports on this FCN.</td>
</tr>
<tr>
<td>Discovered HBA</td>
<td>FibreChannelNode: InferredElement: Inference: Primary: DiscoveredHBA</td>
<td>Primary inference.</td>
</tr>
</tbody>
</table>

FCP node attributes

The attributes and relationships on an FCP node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port Name</td>
<td>name string</td>
<td>Port Name.</td>
</tr>
<tr>
<td>WWPN</td>
<td>wwpn string</td>
<td>World Wide Port Name. The name of the port on the FCN.</td>
</tr>
<tr>
<td>Type</td>
<td>port_type string</td>
<td>Port type.</td>
</tr>
<tr>
<td>State</td>
<td>port_state string</td>
<td>Port state.</td>
</tr>
<tr>
<td>Speed</td>
<td>speed int</td>
<td>Port speed.</td>
</tr>
<tr>
<td>Supported Speeds</td>
<td>supported_speeds list:int</td>
<td>Supported speeds.</td>
</tr>
<tr>
<td>Fabric Name</td>
<td>fabric_name string</td>
<td>Fabric name.</td>
</tr>
<tr>
<td>Supported Classes</td>
<td>supported_classes list:string</td>
<td>Supported classes.</td>
</tr>
<tr>
<td>Role</td>
<td>role string</td>
<td>Port role.</td>
</tr>
</tbody>
</table>

FCP node relationships

The relationships of an FCP node are described in the table below.
**Destination** | **Relationship** | **Description**
---|---|---
Fibre Channel node | FibreChannelPort: FibreChannelPort: FibreChannelNodePort: FibreChannelNodeWithPort: FibreChannelNode | The WWNN for this WWPN.
Discovered HBA | FibreChannelPort: InferredElement: Inference: Primary: DiscoveredHBA | Primary inference.
Connected Storage | FibreChannelPort: Consumer:SANStorage: Producer: StorageConnection | SAN storage connected to a port.
Storage Processor | FibreChannelPort: Contained: Containment: Container: StorageProcessor | Storage processor for this port.
Network Device | FibreChannelPort: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: NetworkDevice | Network device for this port.
Exposed Volumes | FibreChannelPort: FrontEndPort: ExposedView: ExposedElement: StorageVolume | Volumes exposed via this port.

**Network Device node**

The Network Device node represents a network device such as a switch or router. Network Device information is discovered using SNMP. Alternatively, device information could be manually imported.

**Network Device node lifecycle**

The following section describes the scenarios in which Network Device nodes are created, updated or destroyed.

**Creation of discovered Network Devices**

Discovered Network Device nodes are created in exactly the same way as Host nodes.

**Creation of imported Network Devices**

Network Device nodes are created in the datastore when an import script is run, for example the Edge Switch import tool.

A new Network Device node is created if there is no node representing a switch with that name in the datastore.

**Update of discovered Network Devices**

Network Device nodes are updated in exactly the same way as Host nodes.

**Update of imported Network Devices**

A Network Device node is updated in the datastore when an import script is run, for example the CiscoWorks tool. The primary key for a Network Device node is the name. Network Device nodes are updated if a switch in the input data has the same name.
Removal of discovered Network Devices

Network Device nodes are removed in exactly the same way as Host nodes.

Removal of imported Network Devices

A Network Device node can be destroyed from the datastore by user intervention through the User Interface. If a switch is destroyed from the input data, and the import tool is run, the existing Network Device node representing that switch remains in the datastore.

⚠️ There is an option on the Edge Switch import tool to destroy any existing Network Device nodes that are not included in the new import.

Network Device node attributes

The attributes and relationships on a Network Device node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of device.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Model of device.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Vendor of device.</td>
</tr>
<tr>
<td>System Name</td>
<td>sysname string</td>
<td>System name.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>sysobjectid string</td>
<td>System object id.</td>
</tr>
<tr>
<td>Serial ID</td>
<td>serial string</td>
<td>Serial id.</td>
</tr>
<tr>
<td>Capability IDs</td>
<td>capability_ids list:int</td>
<td>Capability type ids of device.</td>
</tr>
<tr>
<td>Capability</td>
<td>capability_types list:string</td>
<td>Capability types of device.</td>
</tr>
<tr>
<td>Device Type</td>
<td>device_type string</td>
<td>Type of device. This attribute is deprecated, use type instead.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of device.</td>
</tr>
<tr>
<td>OS Type</td>
<td>os_type string</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>OS Class</td>
<td>os_class string</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version string</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>os_vendor string</td>
<td>Operating system vendor.</td>
</tr>
<tr>
<td>Testing Status</td>
<td>testing_status -string_</td>
<td>Device testing status</td>
</tr>
<tr>
<td>Nexus VDC ID</td>
<td>nexus_vdc_id string</td>
<td>Cisco Nexus VDC ID.</td>
</tr>
<tr>
<td>Validation Status</td>
<td>validation_status string</td>
<td>Validation status.</td>
</tr>
<tr>
<td>__all_ip_addrs</td>
<td>list:string</td>
<td>Internal attribute to aid searching Network Devices by IP address, these can include IPs from ports, the device and scanned IPs and no specific behavior of this field should be relied upon.</td>
</tr>
<tr>
<td>__all_dns_names</td>
<td>list:string</td>
<td>Internal attribute to aid searching Network Devices by name.</td>
</tr>
<tr>
<td>__cdp_device_id</td>
<td>string</td>
<td>Internal attribute to store SNMP CDP device id.</td>
</tr>
<tr>
<td>age_count</td>
<td>int</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
</tr>
<tr>
<td>last_update_success</td>
<td>date</td>
<td>The time at which a scan was last successfully associated with this Network Device.</td>
</tr>
<tr>
<td>last_update_failure</td>
<td>date</td>
<td>The time at which a scan associated with this Network Device failed.</td>
</tr>
<tr>
<td>last_cmdb_sync_success</td>
<td>date</td>
<td>The time at which this Network Device was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>last_cmdb_sync_failure</td>
<td>date</td>
<td>The time at which an attempt to synchronize this Network Device into the CMDB failed.</td>
</tr>
<tr>
<td>last_cmdb_sync_duration</td>
<td>float</td>
<td>The time in seconds spent performing the last CMDB synchronization of this Network Device.</td>
</tr>
<tr>
<td>last_cmdb_sync_ci_count</td>
<td>int</td>
<td>The number of CIs corresponding to this Network Device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>last_cmdb_sync_rel_count</td>
<td>int</td>
<td>The number of relationships between CIs corresponding to this Network Device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Primary name.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name.</td>
</tr>
</tbody>
</table>

Description of the element.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Attribute Name and Type</strong></td>
<td><strong>Type</strong></td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms list:string</td>
</tr>
</tbody>
</table>

**Network Device node relationships**

The relationships on a Network Device node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discovery Access</strong></td>
<td>NetworkDevice: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this Network Device.</td>
</tr>
<tr>
<td><strong>Not displayed in UI</strong></td>
<td>NetworkDevice: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this Network Device.</td>
</tr>
<tr>
<td><strong>Endpoint Used To Be Identified As</strong></td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td><strong>Endpoint Used To Be Identified As</strong></td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td><strong>Endpoint Now Identified As</strong></td>
<td>NetworkDevice: Previous: EndpointIdentity: Next: Host</td>
<td>Next Host Identity.</td>
</tr>
<tr>
<td><strong>Endpoint Used To Be Identified As</strong></td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td><strong>Endpoint Now Identified As</strong></td>
<td>NetworkDevice: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Next Mainframe Identity.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: Printer</td>
<td>Next Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>NetworkDevice: Previous: EndpointIdentity: Next: Printer</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td>Next SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>NetworkDevice: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>NetworkDevice: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Next Storage Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>NetworkDevice: Previous: EndpointIdentity: Next: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Chosen Endpoint</td>
<td>NetworkDevice: Device: ChosenEndpoint: Endpoint</td>
<td>Endpoint used to discover this NetworkDevice.</td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>NetworkDevice: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: NetworkInterface</td>
<td>Network interfaces of this Network Device.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>NetworkDevice: DeviceWithAddress: DeviceAddress: IPv4Address: IPAddress</td>
<td>IPv4 addresses of this Network Device.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>NetworkDevice: DeviceWithAddress: DeviceAddress: IPv6Address: IPAddress</td>
<td>IPv6 addresses of this Network Device.</td>
</tr>
<tr>
<td>Fibre Channel Ports</td>
<td>NetworkDevice:DeviceWithInterface: DeviceInterface:InterfaceOfDevice: FibreChannelPort</td>
<td>FC ports of this NetworkDevice.</td>
</tr>
<tr>
<td>Subnets</td>
<td>NetworkDevice: DeviceOnSubnet: DeviceSubnet: Subnet: Subnet</td>
<td>The Subnets that this Network Device connects to.</td>
</tr>
</tbody>
</table>
Printed node

A Printer node represents a discovered printer.

Printed node lifecycle

The following section describes the scenarios in which a Printer node is created, updated, or removed.

Creation/update

Printer nodes are created and updated in exactly the same way as Network Device nodes.

Removal

Printer nodes are removed in exactly the same way as Network Device nodes.

Printer node attributes

The attributes and relationships on a Printer node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of printer.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of printer. This attribute was called <code>device_type</code> in BMC Atrium Discovery releases prior to version 9.0. The <code>type</code> attribute should now be used.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Model</td>
<td>Model of printer.</td>
</tr>
<tr>
<td>model</td>
<td>string</td>
</tr>
<tr>
<td>Vendor</td>
<td>Vendor of printer.</td>
</tr>
<tr>
<td>vendor</td>
<td>string</td>
</tr>
<tr>
<td>System Name</td>
<td>System name.</td>
</tr>
<tr>
<td>sysname</td>
<td>string</td>
</tr>
<tr>
<td>System Object ID</td>
<td>System object id.</td>
</tr>
<tr>
<td>sysobjectid</td>
<td>string</td>
</tr>
<tr>
<td>Serial ID</td>
<td>Serial id.</td>
</tr>
<tr>
<td>serial</td>
<td>string</td>
</tr>
<tr>
<td>OS Type</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>os_type</td>
<td>string</td>
</tr>
<tr>
<td>OS Class</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>os_class</td>
<td>string</td>
</tr>
<tr>
<td>OS Version</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>os_version</td>
<td>string</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>Operating system vendor.</td>
</tr>
<tr>
<td>os_vendor</td>
<td>string</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>Firmware version.</td>
</tr>
<tr>
<td>firmware_version</td>
<td>string</td>
</tr>
<tr>
<td>Testing Status</td>
<td>Device testing status.</td>
</tr>
<tr>
<td>testing_status</td>
<td>string</td>
</tr>
<tr>
<td>Capability IDs</td>
<td>Capability type ids of device.</td>
</tr>
<tr>
<td>_capability_ids</td>
<td>list _int</td>
</tr>
<tr>
<td>Capability</td>
<td>Capability types of device.</td>
</tr>
<tr>
<td>capability_types</td>
<td>list string</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by IP address.</td>
</tr>
<tr>
<td>__all_ip_addrs</td>
<td>list:string</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by name.</td>
</tr>
<tr>
<td>__all_dns_names</td>
<td>list:string</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
</tr>
<tr>
<td>age_count_int_</td>
<td>list:int</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The time at which a scan was last successfully associated with this Printer.</td>
</tr>
<tr>
<td>last_update_success</td>
<td>date</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The time at which a scan associated with this Printer failed.</td>
</tr>
<tr>
<td>last_update_failure</td>
<td>date</td>
</tr>
<tr>
<td>Last successful CMDB sync</td>
<td>The time at which this Printer was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>last_cmdb_sync_success</td>
<td>date</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time at which an attempt to synchronize this Printer into the CMDB failed.</td>
</tr>
<tr>
<td>last_cmdb_sync_failure</td>
<td>date</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>last_cmdb_sync_duration_float_</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>last_cmdb_sync_ci_count_int_</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>last_cmdb_sync_rel_count_int_</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms string</td>
</tr>
</tbody>
</table>

**Printer node relationships**

The relationships on a Printer node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>Failure</td>
<td>The failed access for an endpoint.</td>
</tr>
<tr>
<td>Discovery Access</td>
<td>Optimization</td>
<td>The optimized access for an endpoint.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>DeviceInfo</td>
<td>DeviceInfo of this Printer.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Previous: EndpointIdentity: Next: NetworkDevice</td>
<td>Next Network Device Identity.</td>
</tr>
<tr>
<td>Destination Used To Be Identified As</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: Host</td>
<td>Next Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Next Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Next: EndpointIdentity: Previous: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: Printer</td>
<td>Next Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Next SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>Printer: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: StorageDevice</td>
<td>Endpoin used to discover this Printer.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>Printer: Previous: EndpointIdentity: Next: ManagementController</td>
<td>Endpoin used to discover this Printer.</td>
</tr>
<tr>
<td>Chosen Endpoint</td>
<td>Printer: Device: ChosenEndpoint: Endpoint: Endpoin</td>
<td>Endpoin used to discover this Printer.</td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>Printer: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: NetworkInterface</td>
<td>Network interfaces on this Printer.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>Printer: DeviceWithAddress: DeviceAddress: IPv4Address: IPAddress</td>
<td>IPv4 addresses of this Printer.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>Printer: DeviceWithAddress: DeviceAddress: IPv6Address: IPAddress</td>
<td>IPv6 addresses of this Printer.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Printer: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>Printer: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>Printer: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
</tbody>
</table>
SNMP Managed Device node

SNMP Managed Device node
An SNMP Managed Device node represents a device that BMC Atrium Discovery does not have sufficient knowledge of to fully discover. It might be a switch, router, or SNMP enabled coffee machine for which we have not yet implemented support.

SNMP Managed Device node lifecycle
The following section describes the scenarios in which a SNMP Managed Device node is created, updated, or removed.

Creation/update
SNMP Managed Device nodes are created and updated in exactly the same way as Network Device nodes.

Removal
SNMP Managed Device nodes are removed in exactly the same way as Network Device nodes.

SNMP Managed Device node attributes
The attributes and relationships on a SNMP Managed Device node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of device.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of device.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model of device.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>Vendor of device.</td>
<td></td>
</tr>
<tr>
<td>Vendor</td>
<td>Vendor string</td>
<td></td>
</tr>
<tr>
<td>System Name</td>
<td>System name.</td>
<td></td>
</tr>
<tr>
<td>System Object ID</td>
<td>System object id.</td>
<td></td>
</tr>
<tr>
<td>Serial ID</td>
<td>Serial id.</td>
<td></td>
</tr>
<tr>
<td>Serial ID</td>
<td>Serial string</td>
<td></td>
</tr>
<tr>
<td>OS Type</td>
<td>Operating system type.</td>
<td></td>
</tr>
<tr>
<td>OS Class</td>
<td>Operating system class.</td>
<td></td>
</tr>
<tr>
<td>OS Version</td>
<td>Operating system version.</td>
<td></td>
</tr>
<tr>
<td>OS Vendor</td>
<td>Operating system vendor.</td>
<td></td>
</tr>
<tr>
<td>Firmware Version</td>
<td>Firmware version.</td>
<td></td>
</tr>
<tr>
<td>Testing Status</td>
<td>Device testing status.</td>
<td></td>
</tr>
<tr>
<td>Capability IDs</td>
<td>Capability type ids of device</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by IP address.</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by name.</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The time at which a scan was last successfully associated with this device.</td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>The time at which a scan associated with this device failed.</td>
<td></td>
</tr>
<tr>
<td>Last successful CMDB sync</td>
<td>The time at which this device was last successfully synchronized into the CMDB.</td>
<td></td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>The time at which an attempt to synchronize this device into the CMDB failed.</td>
<td></td>
</tr>
<tr>
<td>Last CMDB sync</td>
<td>The time at which this device was last CMDB synchronized into the CMDB.</td>
<td></td>
</tr>
<tr>
<td>Last update success date</td>
<td>The time at which this device was last successfully synchronized into the CMDB.</td>
<td></td>
</tr>
<tr>
<td>Last update failure date</td>
<td>The time at which an attempt to synchronize this device into the CMDB failed.</td>
<td></td>
</tr>
<tr>
<td>Last CMDB sync</td>
<td>The time at which this device was last CMDB synchronized into the CMDB.</td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>The time in seconds spent performing the last CMDB synchronization of this device.</td>
<td></td>
</tr>
<tr>
<td>last_cmdb_sync_duration_float_</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>The number of CIs corresponding to this device at the last CMDB synchronization.</td>
<td></td>
</tr>
<tr>
<td>last_cmdb_sync_ci_count_int_</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>The number of relationships between CIs corresponding to this device at the last CMDB synchronization.</td>
<td></td>
</tr>
<tr>
<td>last_cmdb_sync_rel_count_int_</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>Globally unique key.</td>
<td></td>
</tr>
<tr>
<td>key string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Primary name.</td>
<td></td>
</tr>
<tr>
<td>name string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Short Name</td>
<td>Short name.</td>
<td></td>
</tr>
<tr>
<td>short_name string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Description of the device. Legacy attribute not currently used. Can be used by patterns if desired.</td>
<td></td>
</tr>
<tr>
<td>description string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>URL</td>
<td>URL for information about the device. Legacy attribute not currently used. Can be used by patterns if desired.</td>
<td></td>
</tr>
<tr>
<td>url string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>If true, the device is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
<td></td>
</tr>
<tr>
<td>business_continuity_critical boolean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>If true, the device is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
<td></td>
</tr>
<tr>
<td>third_party_support boolean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Synonyms</td>
<td>Other names by which this device is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
<td></td>
</tr>
<tr>
<td>synonyms string</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SNMP Managed Device node relationships**

The relationships on a SNMP Managed Device node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>SNMPManagedDevice: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this device.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SNMPManagedDevice: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this device.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td></td>
<td>Next Network Device Identity.</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: NetworkDevice</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: Host</td>
<td>Next Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Next Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: Printer</td>
<td>Next Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Next SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>SNMPManagedDevice: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>SNMPManagedDevice: Previous: EndpointIdentity: Next: StorageDevice</td>
<td>Next Storage Device Identity.</td>
</tr>
<tr>
<td>Chosen Endpoint</td>
<td>SNMPManagedDevice: Device: ChosenEndpoint: Endpoint: Endpoint</td>
<td>Endpoint used to discover this device.</td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>SNMPManagedDevice: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: NetworkInterface</td>
<td>Network interfaces on this SNMP Managed Device.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>SNMPManagedDevice: DeviceWithAddress: DeviceAddress: IPv4Address: IPAddress</td>
<td>IPv4 addresses of this SNMP Managed Device.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>SNMPManagedDevice: DeviceWithAddress: DeviceAddress: IPv6Address: IPAddress</td>
<td>IPv6 addresses of this SNMP Managed Device.</td>
</tr>
<tr>
<td>Managed Host</td>
<td>SNMPManagedDevice: Manager: Management: ManagedElement: Host</td>
<td>Host that this SNMP Managed Device manages.</td>
</tr>
<tr>
<td>Container</td>
<td>SNMPManagedDevice: ContainedHost: HostContainment: HostContainer: HostContainer</td>
<td>Host container of which this is a member.</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Managed Host Container</td>
<td>SNMPManagedDevice: Manager: Management: ManagedElement: HostContainer</td>
<td>Host Container that this SNMP Managed Device manages.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>SNMPManagedDevice: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>SNMPManagedDevice: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>SNMPManagedDevice: ElementInCategory: ElementCategory: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Family</td>
<td>SNMPManagedDevice: ElementInCategory: ElementCategory: Family</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>SNMPManagedDevice: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>SNMPManagedDevice: ElementInLocation: Location: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>SNMPManagedDevice: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>SNMPManagedDevice: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>SNMPManagedDevice: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

**Management Controller node**

A Management Controller node is used to model a device integrated into a larger device which provides remote control and information about the larger device, for example, iDRAC, iLOM, and CIMC.

**Management Controller Lifecycle**

The following section describes the scenarios in which a Management Controller is created, updated or removed.

**Creation**

Management Controller nodes are created in exactly the same way as Host nodes.

**Update**

Management Controller nodes are updated in exactly the same way as Host nodes.

**Removal**

Management Controller nodes are removed in exactly the same way as Host nodes.

**Management Controller node attributes**

The attributes and relationships of a Management Controller node are described in the table above.
<table>
<thead>
<tr>
<th>Name</th>
<th>Name of device.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Type of device.</td>
</tr>
<tr>
<td>Model</td>
<td>Model of device.</td>
</tr>
<tr>
<td>Vendor</td>
<td>Vendor of device.</td>
</tr>
<tr>
<td>Management URL</td>
<td>Management URL.</td>
</tr>
<tr>
<td>System Name</td>
<td>System name.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>System object id.</td>
</tr>
<tr>
<td>Serial ID</td>
<td>Serial id.</td>
</tr>
<tr>
<td>OS Type</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>OS Class</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>OS Version</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>Operating system vendor.</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>Firmware version.</td>
</tr>
<tr>
<td>Testing Status</td>
<td>Device testing status.</td>
</tr>
<tr>
<td>Managed System FQDN</td>
<td>FQDN of the system this controller manages.</td>
</tr>
<tr>
<td>Managed System OS</td>
<td>OS of the system this controller manages.</td>
</tr>
<tr>
<td>Managed System Serial ID</td>
<td>Serial id of the system this controller manages.</td>
</tr>
<tr>
<td>Managed System UUID</td>
<td>UUID of the system this controller manages.</td>
</tr>
<tr>
<td>Capability IDs</td>
<td>Capability type ids of device.</td>
</tr>
<tr>
<td>Capability</td>
<td>Capability types of device.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by IP address.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by name.</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td>_all_dns_names</td>
<td>list:string</td>
</tr>
<tr>
<td>age_count</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
</tr>
<tr>
<td>last_update_success</td>
<td>The time at which a scan was last successfully associated with this device.</td>
</tr>
<tr>
<td>last_update_failure</td>
<td>The time at which a scan associated with this device failed.</td>
</tr>
<tr>
<td>last_cmdb_sync_success</td>
<td>The time at which this device was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>last_cmdb_sync_failure</td>
<td>The time at which an attempt to synchronize this device into the CMDB failed.</td>
</tr>
<tr>
<td>last_cmdb_sync_duration</td>
<td>The time in seconds spent performing the last CMDB synchronization of this device.</td>
</tr>
<tr>
<td>last_cmdb_sync_ci_count</td>
<td>The number of CIs corresponding to this device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>last_cmdb_sync_rel_count</td>
<td>The number of relationships between CIs corresponding to this device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>Key</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>key</td>
<td>string</td>
</tr>
<tr>
<td>Name</td>
<td>Primary name.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
</tr>
<tr>
<td>Short Name</td>
<td>Short name.</td>
</tr>
<tr>
<td>short_name</td>
<td>string</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the element.</td>
</tr>
<tr>
<td>description</td>
<td>string</td>
</tr>
<tr>
<td>URL</td>
<td>URL for information about the element.</td>
</tr>
<tr>
<td>url</td>
<td>string</td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>If true, element is critical to operation of the business.</td>
</tr>
<tr>
<td>business_continuity_critical</td>
<td>boolean</td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>True if the element is supported by a third party.</td>
</tr>
<tr>
<td>third_party_support</td>
<td>boolean</td>
</tr>
<tr>
<td>Synonyms</td>
<td>Other names for the element.</td>
</tr>
<tr>
<td>synonyms</td>
<td>list:string</td>
</tr>
</tbody>
</table>

**Management Controller node relationships**

The relationships on a Management Controller node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>ManagementController: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this device.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>--------------------------------------------------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>ManagementController: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this device.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td>Previous SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>ManagementController: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
</tbody>
</table>

Endpoint used to discover this device.
## Management Controller node display relationships

The display relationships on a Management Controller node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chosen Endpoint</td>
<td>ManagementController:: Device:: ChosenEndpoint:: Endpoint:: Endpoint</td>
<td></td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>ManagementController:: DeviceWithInterface:: DeviceInterface:: InterfaceOfDevice:: NetworkInterface</td>
<td>Network interfaces of this device.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>ManagementController:: DeviceWithAddress:: DeviceAddress:: IPv4Address:: IPAddress</td>
<td>IPv4 addresses of this device.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>ManagementController:: DeviceWithAddress:: DeviceAddress:: IPv6Address:: IPAddress</td>
<td>IPv6 addresses of this device.</td>
</tr>
<tr>
<td>Managed Host</td>
<td>ManagementController:: Manager:: Management:: ManagedElement:: Host</td>
<td>Host that this Management Controller manages.</td>
</tr>
<tr>
<td>Container</td>
<td>ManagementController:: Contained:: HostContainment:: HostContainer:: HostContainer</td>
<td>Host container of which this is a member.</td>
</tr>
<tr>
<td>Managed Host Container</td>
<td>ManagementController:: Manager:: Management:: ManagedElement:: HostContainer</td>
<td>Host Container that this Management Controller manages.</td>
</tr>
<tr>
<td>Not displayed in UI.</td>
<td>ManagementController:: AttachmentContainer:: Attachment:: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>ManagementController:: ElementInCategory:: ElementCategory:: Category:: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>ManagementController:: ElementInCategory:: ElementCategory:: Category:: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Family</td>
<td>ManagementController:: ElementInCategory:: ElementCategory:: Category:: Family</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>ManagementController:: OwnedItem:: Ownership:: Owner:: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>ManagementController:: ElementInLocation:: Location:: Location:: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>ManagementController:: OwnedItem:: Ownership:: SupportOwner:: Person</td>
<td>The person or owner responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>ManagementController:: OwnedItem:: Ownership:: BusinessOwner:: Person</td>
<td>The person or owner responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>ManagementController:: OwnedItem:: Ownership:: ITOwner:: Person</td>
<td>The person or owner responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>
Load Balancer nodes

Load balancer information is discovered using SNMP. When BMC Atrium Discovery finds a hardware or virtualized Application Delivery Controller (ADC) that has load balancer options enabled, it creates a network device node and triggers the load balancer discovery TPL patterns to discover the related load balancing components, including pools, hosts and services. When these elements are discovered, the following nodes are modeled:

- Load Balancer Group node
- Load Balancer Instance node
- Load Balancer Member node
- Load Balancer Pool node
- Load Balancer Service node

Load Balancer Group node

The Load Balancer Group node represents a pair or a cluster of load balancers, and links to one or more LoadBalancerInstance nodes. A LoadBalancerGroup is created when BMC Atrium Discovery scans paired/clustered network devices with load balancing options enabled.

Creation/update

A LoadBalancerGroup node is created or updated when BMC Atrium Discovery scans paired/clustered network devices with load balancing options enabled.

Removal

A LoadBalancerGroup node is removed in the following circumstances:

- when the Network Device(s) that was scanned to create it is removed
- when all the LoadBalancerInstance nodes linked to it are removed
- when a subsequent scan detects that the corresponding load balancer instances are no longer paired or clustered

Load Balancer Group node attributes

The attributes of a Load Balancer Group node are described in the table below.
Load Balancer Group node relationships
The relationships for a Load Balancer Group node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load Balancers</td>
<td>LoadBalancerGroup:Container:Containment:ContainedInstance:LoadBalancerInstance</td>
<td>Load balancer instances that are part of this group.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>LoadBalancerGroup:Element:Maintainer:Pattern:Pattern</td>
<td>Pattern that is maintaining this load balancer group.</td>
</tr>
</tbody>
</table>

Load Balancer Instance node
The Load Balancer Instance node represents a configured load balancer instance and is created when BMC Atrium Discovery scans a network device with load balancing options enabled.

Creation/update
A Load Balancer Instance node is created or updated when BMC Atrium Discovery scans a network device with load balancing options enabled.

Removal
A Load Balancer Instance node is removed when the Network Device that was scanned to create it is removed.

When a Load Balancer Instance node is removed, all of the nodes it contains are removed. This is Containment Removal, see Containment Removal.

Load Balancer Instance node attributes
The attributes of a Load Balancer Instance node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key key</td>
<td>string</td>
<td>Globally unique key of the load balancer instance.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>The type of load balancer instance.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td><code>type string</code></td>
<td>Name of the load balancer instance.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td><code>name string</code></td>
<td>Version of the load balancer instance.</td>
</tr>
<tr>
<td><strong>Failover Type</strong></td>
<td><code>failover_type string</code></td>
<td>Type of failover being used (for example, Active-Standby, Active-Active).</td>
</tr>
<tr>
<td><strong>Failover State</strong></td>
<td><code>failover_state string</code></td>
<td>State of the failover member (for example, Active or Standby).</td>
</tr>
<tr>
<td><strong>Failover Unit</strong></td>
<td><code>failover_unit_id int</code></td>
<td>Failover unit id.</td>
</tr>
</tbody>
</table>

### Load Balancer Instance node relationships

The relationships for a Load Balancer Instance node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device</td>
<td><code>LoadBalancerInstance:NetworkService:NetworkService:NetworkDevice:</code></td>
<td>Device that the load balancer is running on.</td>
</tr>
<tr>
<td>Services</td>
<td><code>LoadBalancerInstance:Container:Containment:ContainedService:</code></td>
<td>Services that are being load balanced.</td>
</tr>
<tr>
<td>Failover Group</td>
<td><code>LoadBalancerInstance:ContainedInstance:Containment:Container:</code></td>
<td>Load balancer group that this instance is part of.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td><code>LoadBalancerInstance:Element:Maintainer:Pattern:</code></td>
<td>Pattern that is maintaining this load balancer instance.</td>
</tr>
</tbody>
</table>

### Load Balancer Member node

The Load Balancer Member node represents a physical server that acts as a provider of a service available to a load balancer.
Creation/update
A Load Balancer Member node is created or updated when BMC Atrium Discovery scans a network device with load balancing options enabled.

Removal
A Load Balancer Member node is removed when its parent Load Balancer Instance node is removed. This is Containment Removal, see Containment Removal.

Load Balancer Member node attributes
The attributes of a Load Balancer Member node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key of the LoadBalancerService.</td>
</tr>
<tr>
<td>IP Address</td>
<td>ip_addr string</td>
<td>IP address of the service.</td>
</tr>
<tr>
<td>Port</td>
<td>port string</td>
<td>Port that the service is using.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of the service.</td>
</tr>
<tr>
<td>State</td>
<td>state string</td>
<td>State of the service.</td>
</tr>
</tbody>
</table>

Load Balancer Member node relationships
The relationships for a Load Balancer Member node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pools</td>
<td>LoadBalancerMember:ContainedMember:Containment:Container:LoadBalancerPool</td>
<td>Pools that the Load Balancer Member is in.</td>
</tr>
<tr>
<td>Host</td>
<td>LoadBalancerMember:Service:SoftwareService:ServiceHost:Host</td>
<td>Host implementing the service.</td>
</tr>
<tr>
<td>Software Instance</td>
<td>LoadBalancerMember:Service:SoftwareService:ServiceProvider:SoftwareInstance</td>
<td>Software Instance implementing the service.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Load Balancer Pool node

The Load Balancer Pool node represents a pool of members on a load balancer.

Creation/update

A Load Balancer Pool node is created or updated when BMC Atrium Discovery scans a network device with load balancing options enabled.

Removal

A Load Balancer Pool node is removed when its parent Load Balancer Instance node is removed. This is Containment Removal, see Containment Removal.

Load Balancer Pool node attributes

The attributes of a Load Balancer Pool node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key of the LoadBalancerPool.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of the pool.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the pool.</td>
</tr>
<tr>
<td>Protocol</td>
<td>protocol string</td>
<td>Protocol of the pool.</td>
</tr>
<tr>
<td>Algorithm</td>
<td>algorithm string</td>
<td>Load balancing algorithm used by the pool.</td>
</tr>
<tr>
<td>NAT Type</td>
<td>nat_type string</td>
<td>Type of NAT employed by members of the pool.</td>
</tr>
</tbody>
</table>

Load Balancer Pool node relationships

The relationships for a Load Balancer Pool node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load balancer</td>
<td>LoadBalancerPool:ContainedPool:Containment:Container:LoadBalancerInstance</td>
<td>The load balancer where the pool is configured.</td>
</tr>
</tbody>
</table>
### Load Balancer Service node

The Load Balancer Service node represents a virtual server on a load balancer.

#### Creation/update

A Load Balancer Service node is created or updated when BMC Atrium Discovery scans a network device with load balancing options enabled.

#### Removal

A Load Balancer Service node is removed when its parent Load Balancer Instance node is removed. This is Containment Removal, see [Containment Removal](#).

#### Load Balancer Service node attributes

The attributes of a Load Balancer Service node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key key string</td>
<td>Globally unique key of the LoadBalancerService.</td>
</tr>
<tr>
<td>Name name string</td>
<td>Name of the LoadBalancerService.</td>
</tr>
<tr>
<td>IP Address ip_addr string</td>
<td>IP address of the LoadBalancerService.</td>
</tr>
<tr>
<td>Port port string</td>
<td>Port of the LoadBalancerService.</td>
</tr>
<tr>
<td>Not displayed in UI enabled string enabled string</td>
<td>Whether LoadBalancerService is enabled.</td>
</tr>
<tr>
<td>Type type string</td>
<td>Type of LoadBalancerService.</td>
</tr>
<tr>
<td></td>
<td>State of LoadBalancerService.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>State</td>
<td>state string</td>
</tr>
<tr>
<td>DNS Names</td>
<td>dns_names list:string</td>
</tr>
</tbody>
</table>

Load Balancer Service node relationships

The relationships for a LoadBalancerService node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load balancer</td>
<td>LoadBalancerService:ContainedService:Containment:Container:LoadBalancerInstance</td>
<td>The instance to which this service is linked to.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>LoadBalancerService:Element:Maintainer:Pattern:Pattern</td>
<td>Pattern that is maintaining this load balancer service</td>
</tr>
</tbody>
</table>

IP Address node

IP Address node
An IP Address node represents an IPv4 or IPv6 address of a host or a device.

IP Address node lifecycle
The following section describes the scenarios in which an IP Address node is created, updated or destroyed.

Creation
An IP Address node is created in the datastore when a host, network device, printer, or SNMP managed device is scanned successfully. The new IP Address node will only be created if one with the same key does not already exist as a child of that node.

Update
An IP Address node is updated every time the parent host or a device is scanned successfully with sufficient credentials to run the discovery command.
Removal

To remove an IP Address node manually, find the necessary IP Address node, select it in the list and pick **Destroy** from the **Actions** list.

An IP Address node is automatically destroyed if it is not seen on a subsequent scan of its parent host or device. This is an Authoritative Removal type, see **Authoritative Removal**. The IP Address node is also destroyed when its parent host or device is destroyed. This is a Containment Removal type because an IP Address cannot exist without its parent host or device, see **Containment Removal**. For information on the destruction of Host nodes, see **Host node lifecycle**.

**IP Address node attributes**

The attributes and relationships on an IP Address node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI key string</td>
<td>Unique key.</td>
<td></td>
</tr>
<tr>
<td>Name name string</td>
<td>Name of the IP Address. This is of the form IP number on host.</td>
<td></td>
</tr>
<tr>
<td>Type type string</td>
<td>IP address type (IPv4 or IPv6).</td>
<td></td>
</tr>
<tr>
<td>IP Address ip_addr string</td>
<td>IP address in canonical form.</td>
<td></td>
</tr>
<tr>
<td>Broadcast Address broadcast string</td>
<td>IPv4 broadcast address.</td>
<td></td>
</tr>
<tr>
<td>Netmask netmask string</td>
<td>IPv4 subnet mask.</td>
<td></td>
</tr>
<tr>
<td>Prefix prefix string</td>
<td>IPv6 prefix in canonical (CIDR) form.</td>
<td></td>
</tr>
<tr>
<td>Prefix Length prefix_length int</td>
<td>IPv6 prefix length in bits.</td>
<td></td>
</tr>
<tr>
<td>Link Local link_local boolean</td>
<td>Whether it is an IPv6 link local address.</td>
<td></td>
</tr>
<tr>
<td>Site Local site_local boolean</td>
<td>Whether it is an IPv6 site local address.</td>
<td></td>
</tr>
<tr>
<td>FQDNs fqdns list:string</td>
<td>Fully Qualified Domain Names.</td>
<td></td>
</tr>
<tr>
<td>Interface Aliases interface_aliases list:string</td>
<td>The interface alias on platforms that report addresses in this way.</td>
<td></td>
</tr>
<tr>
<td>Temporary temporary boolean</td>
<td>Indicates a temporary IPv6 address where this can be detected/reported.</td>
<td></td>
</tr>
<tr>
<td>Zone zone string</td>
<td>The name of the zone this address is in.</td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>server string</td>
<td>Name of the virtual server that uses this address.</td>
</tr>
</tbody>
</table>

**IP Address node relationships**

The relationships on an IP Address node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: Host</td>
<td>The host that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Host</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: Host</td>
<td>The host that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Network Device</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: NetworkDevice</td>
<td>The network device that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Network Device</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: NetworkDevice</td>
<td>The network device that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Printer</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: Printer</td>
<td>The printer that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Printer</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: Printer</td>
<td>The printer that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: SNMPManagedDevice</td>
<td>The SNMP managed device that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: SNMPManagedDevice</td>
<td>The SNMP managed device that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Management Controller</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: ManagementController</td>
<td>The Management Controller that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Management Controller</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: ManagementController</td>
<td>The Management Controller that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Storage Device</td>
<td>IPAddress: IPv4Address: DeviceAddress: DeviceWithAddress: StorageDevice</td>
<td>The storage device that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Storage Device</td>
<td>IPAddress: IPv6Address: DeviceAddress: DeviceWithAddress: StorageDevice</td>
<td>The storage device that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Subnet</td>
<td>IPAddress: DeviceOnSubnet: DeviceSubnet: Subnet: Subnet</td>
<td>The subnet that this IP address belongs to.</td>
</tr>
<tr>
<td>Network Interface</td>
<td>IPAddress: IPv4Address: InterfaceAddress: InterfaceWithAddress: NetworkInterface</td>
<td>The network interfaces that this IPv4 address belongs to.</td>
</tr>
<tr>
<td>Network Interface</td>
<td>IPAddress: IPv6Address: InterfaceAddress: InterfaceWithAddress: NetworkInterface</td>
<td>The network interfaces that this IPv6 address belongs to.</td>
</tr>
<tr>
<td>Discovered IP Address</td>
<td>IPAddress: InferredElement: Inference: Primary: DiscoveredIPAddress</td>
<td>Primary Inference.</td>
</tr>
</tbody>
</table>
Network Interface node

A Network Interface node represents the network interface of an inferred device.

The loopback interface `lo` (IPv4 address `127.0.0.1`, IPv6 address `::1`) is never modeled.

Network Interface node lifecycle

The following section describes the scenarios in which a Network Interface node is created, updated or destroyed.

Creation

A Network Interface node is created in the datastore when a host, network device, printer, or SNMP managed device is scanned successfully. The new Network Interface node will only be created if one with the same key does not already exist as a child of that node.

Update

A Network Interface node is updated every time the parent host or device is scanned successfully with sufficient credentials to run the discovery command.

Removal

To remove a Network Interface node manually, find the necessary Network Interface node, select it in the list and pick **Destroy** from the **Actions** list. A Network Interface node is automatically destroyed if it is not seen on a subsequent scan of its parent host or device. This is an Authoritative Removal type, see [Authoritative Removal](#). The Network Interface node is also destroyed when its parent host or device is destroyed. This is a Containment Removal type because a NetworkInterface cannot exist without its parent host or device, see [Containment Removal](#). For information on the destruction of Host nodes, see [Host node lifecycle](#).

Network Interface node attributes

The attributes and relationships on a Network Interface node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Unique key.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Full name of the network interface.</td>
</tr>
<tr>
<td>Interface Name</td>
<td>interface_name string</td>
<td>Interface name of the network interface.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MAC Address</td>
<td>mac_addr string</td>
<td>Physical (MAC) Address.</td>
</tr>
<tr>
<td>Speed (Mbps)</td>
<td>speed int</td>
<td>Current speed in Mbps.</td>
</tr>
<tr>
<td>Speed (bps)</td>
<td>raw_speed int</td>
<td>Current speed in bps.</td>
</tr>
<tr>
<td>Duplex</td>
<td>duplex string</td>
<td>Duplex setting (Full/Half).</td>
</tr>
<tr>
<td>Negotiation</td>
<td>negotiation string</td>
<td>Negotiation setting.</td>
</tr>
<tr>
<td>Bonded</td>
<td>bonded boolean</td>
<td>A flag which indicates that this interface represents a bonded (teamed/aggregated) interface.</td>
</tr>
<tr>
<td>Aggregation Mode</td>
<td>aggregation_mode string</td>
<td>Shows which mode the aggregation is in.</td>
</tr>
<tr>
<td>Aggregates</td>
<td>aggregates list:string</td>
<td>The interfaces aggregated by this interface.</td>
</tr>
<tr>
<td>Aggregated By</td>
<td>aggregated_by string</td>
<td>The interface which aggregates this interface.</td>
</tr>
<tr>
<td>Aggregated With</td>
<td>aggregated_with list:string</td>
<td>The interfaces aggregated with this interface.</td>
</tr>
<tr>
<td>ifIndex</td>
<td>ifindex int</td>
<td>The ifIndex value.</td>
</tr>
<tr>
<td>Console</td>
<td>is_console boolean</td>
<td>A flag which indicates that this interface represents the service console for an ESX host.</td>
</tr>
<tr>
<td>Interface Type</td>
<td>interface_type string</td>
<td>Type of interface where it is known.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>The SNMP\WMI description for the interface.</td>
</tr>
<tr>
<td>Group Name</td>
<td>group_name string</td>
<td>The name of the IPMP group this interface is in.</td>
</tr>
<tr>
<td>Virtual Adapters</td>
<td>virtual_adapters list:string</td>
<td>The virtual adapters dependent on this physical adapter.</td>
</tr>
<tr>
<td>Physical Adapters</td>
<td>physical_adapters list:string</td>
<td>The physical adapters supporting this virtual adapter.</td>
</tr>
<tr>
<td>Shared Adapters</td>
<td>shared_adapters list:string</td>
<td>The shared adapters this physical or virtual adapter depends on.</td>
</tr>
<tr>
<td>Physical Location</td>
<td>physical_location string</td>
<td>The AIX adapter location.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows: network device ID.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Device ID</td>
<td>The type of this network adapter.</td>
<td></td>
</tr>
<tr>
<td>Adapter Type</td>
<td>The type of this network adapter.</td>
<td></td>
</tr>
<tr>
<td>DHCP Enabled</td>
<td>Indicates if this interface is configured by DHCP.</td>
<td></td>
</tr>
<tr>
<td>Setting GUID</td>
<td>Windows: Setting GUID.</td>
<td></td>
</tr>
<tr>
<td>Service Name</td>
<td>Windows: Service Name.</td>
<td></td>
</tr>
<tr>
<td>Index</td>
<td>Windows: Interface index, used as zoneid for IPv6.</td>
<td></td>
</tr>
<tr>
<td>DNS Hostname</td>
<td>DNS Hostname for this interface.</td>
<td></td>
</tr>
<tr>
<td>Database Path</td>
<td>Windows: path to the network database files.</td>
<td></td>
</tr>
<tr>
<td>DNS Servers</td>
<td>DNS Servers for this interface.</td>
<td></td>
</tr>
<tr>
<td>Default Gateway</td>
<td>Default gateway.</td>
<td></td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Network adapter manufacturer.</td>
<td></td>
</tr>
<tr>
<td>DNS Domain</td>
<td>DNS Domain for this interface.</td>
<td></td>
</tr>
<tr>
<td>DHCP Server</td>
<td>DHCP server for this interface.</td>
<td></td>
</tr>
<tr>
<td>Driver Version</td>
<td>Network driver version.</td>
<td></td>
</tr>
<tr>
<td>Driver Date</td>
<td>Network driver date.</td>
<td></td>
</tr>
<tr>
<td>Primary WINS Server</td>
<td>Primary WINS server for this interface.</td>
<td></td>
</tr>
<tr>
<td>Secondary WINS Server</td>
<td>Secondary WINS server for this interface.</td>
<td></td>
</tr>
<tr>
<td>Index</td>
<td>The index number of the interface.</td>
<td></td>
</tr>
</tbody>
</table>

**Network Interface node relationships**

The relationships on a Network Interface node are described in the table below.
<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: Host</td>
<td>The host that this network interface belongs to.</td>
</tr>
<tr>
<td>Network Device</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: NetworkDevice</td>
<td>The network device that this network interface belongs to.</td>
</tr>
<tr>
<td>Printer</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: Printer</td>
<td>The printer that this network interface belongs to.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: SNMPManagedDevice</td>
<td>The SNMP managed device that this network interface belongs to.</td>
</tr>
<tr>
<td>Management Controller</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: ManagementController</td>
<td>The Management Controller that this network interface belongs to.</td>
</tr>
<tr>
<td>Storage Device</td>
<td>NetworkInterface: InterfaceOfDevice: DeviceInterface: DeviceWithInterface: StorageDevice</td>
<td>The storage device that this network interface belongs to.</td>
</tr>
<tr>
<td>IP Address</td>
<td>NetworkInterface: InterfaceWithAddress: InterfaceAddress: IPv4Address: IPAddress</td>
<td>The IPv4 address of this network interface.</td>
</tr>
<tr>
<td>IP Address</td>
<td>NetworkInterface: InterfaceWithAddress: InterfaceAddress: IPv6Address: IPAddress</td>
<td>The IPv6 address of this network interface.</td>
</tr>
<tr>
<td>Edge Device</td>
<td>NetworkInterface: EdgeClient: NetworkLink: EdgeDevice: NetworkInterface</td>
<td>The edge device interface that this client interface is connected to.</td>
</tr>
<tr>
<td>Edge Client</td>
<td>NetworkInterface: EdgeDevice: NetworkLink: EdgeClient: NetworkInterface</td>
<td>The client interface that this edge device interface is connected to.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>NetworkInterface: InferredElement: Inference: Primary: DiscoveredInterface</td>
<td>Primary Inference.</td>
</tr>
</tbody>
</table>

**Subnet node**

A Subnet node represents an IP subnet on which IP addresses have been scanned.

**Subnet Lifecycle**

The following section describes the scenarios in which a Subnet node is created, updated or destroyed.

**Creation**

A new Subnet node is created in the datastore when a host, network device, printer, or SNMP managed device is scanned and its DiscoveredIPAddress is on a subnet that is not represented in the datastore. Subnet nodes are not created for the IPv6 link local prefix.

The generated key for a Subnet node is based on the IP range.
Update
A Subnet node’s identity and entire data content are the same thing and are consequently not updated.

Removal
To remove a Subnet node manually, find the necessary Subnet node, select it in the list and pick **Destroy** from the **Actions** list.

Subnet Attributes
The attributes and relationships of a Subnet node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Unique key.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>IP address type (IPv4 or IPv6).</td>
</tr>
<tr>
<td>IP range</td>
<td>ip_address_range string</td>
<td>Range of IP addresses in this subnet.</td>
</tr>
</tbody>
</table>

Subnet Relationships
The relationships of a Subnet node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Addresses</td>
<td>Subnet: Subnet: DeviceSubnet:</td>
<td>IP addresses on this subnet.</td>
</tr>
<tr>
<td></td>
<td>DeviceOnSubnet: IPAddress</td>
<td></td>
</tr>
<tr>
<td>Triggering DiscoveredIPAddress</td>
<td>Subnet: InferredElement: Inference: Primary: DiscoveredIPAddress</td>
<td>DiscoveredIPAddress used to infer existence of this subnet.</td>
</tr>
<tr>
<td>Location</td>
<td>Subnet: ElementInLocation: Location: Location</td>
<td>Location of this subnet.</td>
</tr>
</tbody>
</table>

Detail node
A Detail node is a multipurpose node kind that, as its name suggests, is used to store additional details of other nodes. Typically it would be used when there is a need to add something more than some basic attributes to augment existing nodes in the model. If you are extending database Software Instance nodes then you should use a [Database Detail node](#).

Detail nodes have two default relationships defined that allow for multiple levels of modeling. For tree-like structures use the containment relationship `Detail:Contained:Containment: Container:Detail`. For record/table-like structures use the list relationship `Detail:List: List:Member:Detail`. 
By default software instance, business application instance and host nodes all have relationships to detail nodes. This does not prevent you relating detail nodes to other node kinds.

**Detail node lifecycle**

The following section describes the scenarios in which a detail node is created, updated or removed.

**Creation/update**

This is under the full control of patterns and as a result there is no default detail node behavior.

⚠️ **Setting the key for a detail node**

The key for a detail node is entirely dependent on the pattern that creates the detail node. It is advised therefore that you take extra care when constructing the key attribute, as it will need to be unique amongst all detail nodes. Achieving this uniqueness would typically be done by including the following information in the key:

- The `type` attribute
- The parent node’s `key` attribute

**Removal**

- Manual removal: find the necessary detail node, select it in the list and pick **Destroy** from the Actions list.
- Authoritative removal by the pattern that creates/updates the detail node should be considered. The pattern not only needs to create the correct detail structure, it also needs to maintain it as the configuration changes.
- Built in removal rules will remove all the contained detail nodes if an SI/BAI/Host is removed.
- Built in removal rules will remove all the child detail nodes (over both Containment and List) to allow simple deletion of part structures.

**Detail attributes**

The attributes and relationships on a detail node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>name string</code></td>
<td>Name of the detail.</td>
</tr>
<tr>
<td>Short Name</td>
<td><code>short_name string</code></td>
<td>Short name of the detail.</td>
</tr>
</tbody>
</table>

Globally unique key.
**UI Name**

<table>
<thead>
<tr>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key key string</td>
<td></td>
</tr>
<tr>
<td>Type type string</td>
<td>Type of detail.</td>
</tr>
<tr>
<td>Trigger Count count string</td>
<td>Trigger count.</td>
</tr>
</tbody>
</table>

**Detail relationships**

The relationships on a Detail node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Instance</td>
<td>Detail: Detail: Detail: ElementWithDetail: SoftwareInstance</td>
<td>Software Instance with this detail.</td>
</tr>
<tr>
<td>Runtime Environment</td>
<td>Detail: Detail: Detail: ElementWithDetail: RuntimeEnvironment</td>
<td>Runtime Environment with this detail.</td>
</tr>
<tr>
<td>Business Application Instance</td>
<td>Detail: Detail: Detail: ElementWithDetail: BusinessApplicationInstance</td>
<td>Business Application Instance with this detail.</td>
</tr>
<tr>
<td>Host</td>
<td>Detail: Detail: Detail: ElementWithDetail: Host</td>
<td>Host with this detail.</td>
</tr>
<tr>
<td>MFPart</td>
<td>Detail: Detail: Detail: ElementWithDetail: MFPart</td>
<td>MFPart with this detail.</td>
</tr>
<tr>
<td>Element</td>
<td>Detail: Detail: Detail: ElementWithDetail: GenericElement</td>
<td>Element with this detail.</td>
</tr>
<tr>
<td>Detail List</td>
<td>Detail: Member: List: List: Detail</td>
<td>Detail list this Detail is part of.</td>
</tr>
<tr>
<td>Members</td>
<td>Detail: List: List: Member: Detail</td>
<td>List members of this Detail.</td>
</tr>
<tr>
<td>Containing Details</td>
<td>Detail: Contained: Containment: Container: Detail</td>
<td>Detail this Detail is contained within.</td>
</tr>
<tr>
<td>Contents</td>
<td>Detail: Container: Containment: Contained: Detail</td>
<td>Details contained within this Detail.</td>
</tr>
<tr>
<td>Depends On Software Instance</td>
<td>Detail: Dependant: Dependency: DependedUpon: SoftwareInstance</td>
<td>Software Instances that depend upon this Detail.</td>
</tr>
<tr>
<td>Depends Upon By Software Instance</td>
<td>Detail: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instances that this Detail depends on.</td>
</tr>
<tr>
<td>Depends On Software Component</td>
<td>Detail: Dependant: Dependency: DependedUpon: SoftwareComponent</td>
<td>Software Components that depend upon this Detail.</td>
</tr>
<tr>
<td>Depends Upon By Software Component</td>
<td>Detail: DependedUpon: Dependency: Dependant: SoftwareComponent</td>
<td>Software Components that this Detail depends on.</td>
</tr>
<tr>
<td>Depends On Files</td>
<td>Detail: Dependant: Dependency: DependedUpon: File</td>
<td>File depended upon by this Detail.</td>
</tr>
</tbody>
</table>
## Database Detail node

A Database Detail node is a multipurpose node kind that, as its name suggests, is used to store additional details of databases. Typically it would be used when there is a need to add table and schema information for existing database Software Instance nodes.

### Database Detail node lifecycle

The following section describes the scenarios in which a Database Detail node is created, updated or removed.

#### Creation/update

This is under the full control of patterns and as a result there is no default Database Detail node behavior.

#### Setting the key for a Database Detail node

The key for a Database Detail node is entirely dependent on the pattern that creates the Database Detail node. It is advised therefore that you take extra care when constructing the key attribute, as it will need to be unique amongst all Database Detail nodes. Achieving this uniqueness would typically be done by including the following information in the key:

- The `type` attribute
- The parent node's `key` attribute

#### Removal

- Manual removal: find the necessary Detail node, select it in the list and pick **Destroy** from the **Actions** list.
- Authoritative removal by the pattern that creates/updates the Database Detail node should be considered. The pattern not only needs to create the correct Database Detail structure, it also needs to maintain it as the configuration changes.
- Built in removal rules will remove all the contained Database Detail nodes if an SI/BAI/Host is removed.
- Built in removal rules will remove all the child Database Detail nodes (over both Containment and List) to allow simple deletion of part structures.
### Database Detail Attributes

The attributes and relationships on a Database Detail node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of the detail.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name of the detail</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of detail.</td>
</tr>
<tr>
<td>Trigger Count</td>
<td>count string</td>
<td>Trigger count.</td>
</tr>
</tbody>
</table>

### Database Detail Relationships

The relationships on a Database Detail node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Detail List</td>
<td>DatabaseDetail: Member: List: List: DatabaseDetail</td>
<td>Database Detail list this Database Detail is part of.</td>
</tr>
<tr>
<td>Database Members</td>
<td>DatabaseDetail: List: List: Member: DatabaseDetail</td>
<td>List members of this Database Detail.</td>
</tr>
<tr>
<td>Parent Database Details</td>
<td>DatabaseDetail: Contained: Containment: Container: DatabaseDetail</td>
<td>Database Detail this Database Detail is contained within.</td>
</tr>
<tr>
<td>Database Detail Contents</td>
<td>DatabaseDetail: Contained: Containment: Contained: DatabaseDetail</td>
<td>Database Details contained within this Database Detail.</td>
</tr>
<tr>
<td>Containing Database Details</td>
<td>DatabaseDetail: Contained: Containment: Container: DatabaseDetail</td>
<td>Database Detail this Database Detail is contained within.</td>
</tr>
<tr>
<td>Database Contents</td>
<td>DatabaseDetail: Contained: Containment: Contained: DatabaseDetail</td>
<td>Database Details contained within this Database Detail.</td>
</tr>
<tr>
<td>Database Dependant Software Instance</td>
<td>DatabaseDetail: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instance that depends on this Detail.</td>
</tr>
<tr>
<td>Database Dependant Business Application</td>
<td>DatabaseDetail: DependedUpon: Dependency: Dependant: BusinessApplicationInstance</td>
<td>Business Application Instance that depends on this Detail.</td>
</tr>
<tr>
<td>Software Instance</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: SoftwareInstance</td>
<td>Software Instance with this detail.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>Runtime Environment</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: RuntimeEnvironment</td>
<td>Runtime Environment with this detail.</td>
</tr>
<tr>
<td>Business Application Instance</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: BusinessApplicationInstance</td>
<td>Business Application Instance with this detail.</td>
</tr>
<tr>
<td>Host</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: Host</td>
<td>Host with this detail.</td>
</tr>
<tr>
<td>MFPart</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: MFPart</td>
<td>MFPart with this detail.</td>
</tr>
<tr>
<td>Element</td>
<td>DatabaseDetail: Detail: Detail: ElementWithDetail: GenericElement</td>
<td>Element with this detail.</td>
</tr>
<tr>
<td>Detail List</td>
<td>DatabaseDetail: Member: List: List: Detail</td>
<td>Detail list this Detail is part of.</td>
</tr>
<tr>
<td>Members</td>
<td>DatabaseDetail: List: List: Member: Detail</td>
<td>List members of this Detail.</td>
</tr>
<tr>
<td>Containing Details</td>
<td>DatabaseDetail: Contained: Containment: Container: Detail</td>
<td>Detail this Detail is contained within.</td>
</tr>
<tr>
<td>Contents</td>
<td>DatabaseDetail: Container: Containment: Contained: Detail</td>
<td>Details contained within this Detail.</td>
</tr>
<tr>
<td>Depends On Software Instance</td>
<td>DatabaseDetail: Dependant: Dependency: DependedUpon: SoftwareInstance</td>
<td>Software Instances that depend upon this Detail.</td>
</tr>
<tr>
<td>Depended Upon By Software Instance</td>
<td>DatabaseDetail: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instances that this Detail depends on.</td>
</tr>
<tr>
<td>Depends On Software Component</td>
<td>DatabaseDetail: Dependant: Dependency: DependedUpon: SoftwareComponent</td>
<td>Software Components that depend upon this Detail.</td>
</tr>
<tr>
<td>Depended Upon By Software Component</td>
<td>DatabaseDetail: DependedUpon: Dependency: Dependant: SoftwareComponent</td>
<td>Software Components that this Detail depends on.</td>
</tr>
<tr>
<td>Depends On Files</td>
<td>DatabaseDetail: Dependant: Dependency: DependedUpon: File</td>
<td>File depended upon by this Detail.</td>
</tr>
<tr>
<td>Functional Component</td>
<td>DatabaseDetail: ContainerFunctionality: Containment: Container: FunctionalComponent</td>
<td>Functional Components that contain this Detail.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>DatabaseDetail: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this detail.</td>
</tr>
</tbody>
</table>

**Storage node**

A Storage node represents a unit of storage such as an attached tape or disk drive. The type of storage is not mandated by the system; it can be physical, or logical.
This Storage node is not part of the Storage capabilities introduced in BMC Atrium Discovery 10.1. The Storage nodes which represent large scale remote storage devices are described under the umbrella heading Storage nodes.

Storage node lifecycle

The following section describes the scenarios in which a Storage node is created, updated or removed.

Creation/update

If a pattern triggers on a Directly Discovered Data node, such as a Discovered Tape Drive node, it can choose whether to specify keys for the Storage nodes it creates and maintains. If a key is specified then the decision whether to create a new Storage node or to update an existing one depends on the key. If a Storage node with the specified key exists, that node is updated, even if the node was previously maintained by a different pattern. In this case, the pattern takes over as the maintainer of the Storage node. If a node with the specified key does not exist, a new Storage node is created. In both cases, the Storage node is linked to the pattern with a maintainer relationship.

If a key for the Storage node is not specified by the pattern, the system creates or updates a group Storage with an automatically generated key. The key is based upon the key of the Host upon which the Storage is running, the specified type of the Storage and, optionally, a key group that can be used to separate the nodes into a number of groups. The count attribute is set to the number of instances in the group identified in the collection of Directly Discovered Data. Each time the host is scanned, the count attribute is changed to represent the number of instances seen in that scan.

Removal

A Storage node can be destroyed either manually or automatically.

To remove a Storage node manually, find the necessary Storage, select it in the list and pick Destroy from the Actions list.

Automatic removal happens according to the following scenario. The age_count attribute of the Storage node contains information about when the Storage node was last confirmed by its maintaining pattern. If the age_count is positive, it represents the number of consecutive scans of the MFPart node in which the Storage was confirmed. If the age_count is negative, it represents the number of consecutive scans in which the Storage node was not confirmed. The last_update_success and last_update_failure attributes contain the date and time at which the Storage node was last confirmed, and not confirmed, respectively.

The default aging strategy only applies to Storage nodes created from patterns triggering on the following node kinds and maintaining the Storage nodes:

- DiscoveredTapeDrive
DiscoveredDiskDrive

If the Storage is triggered on anything else, then aging must be implemented in the pattern using a removal block.

If the pattern does not have a removal block, Storage nodes are removed using an aging strategy based on the `age_count` and `last_update_success` attributes. The default aging parameters are the same as for a Software Instance node, that is, if a Storage node has not been seen for at least 7 scans, over a period of at least 10 days, it is destroyed.

If the pattern maintaining a node does have a removal block, the block can override the default aging scheme to destroy its nodes either earlier or later than normal. For TKU patterns, refer to the documentation accompanying each pattern for details of special removal behavior.

Regardless of the presence or absence of a removal block in the pattern, if the MFPart corresponding to a DDD-triggered Storage node is destroyed, the Storage node is immediately destroyed (see How nodes get removed).

Storage node attributes

The attributes on a Storage node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td><code>type string</code></td>
<td>Type of Storage.</td>
</tr>
<tr>
<td>Vendor</td>
<td><code>vendor string</code></td>
<td>Vendor.</td>
</tr>
<tr>
<td>Model</td>
<td><code>model string</code></td>
<td>Model.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td><code>device_id string</code></td>
<td>Identifier.</td>
</tr>
<tr>
<td>Name</td>
<td><code>name string</code></td>
<td>The name that the Storage is known by.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td><code>key string</code></td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Description</td>
<td><code>description string</code></td>
<td>Description of the Storage. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td><code>url string</code></td>
<td>URL for information about the Storage. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td><code>business_continuity_critical boolean</code></td>
<td>If true, the Storage is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td><code>third_party_support boolean</code></td>
<td>If true, the Storage is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
</tbody>
</table>
### Storage node relationships

The relationships on a Storage node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used by Host</td>
<td>Storage: Storage: Storage: Host: Host</td>
<td>Host with Storage.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>Storage: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this Storage.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Storage: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>Storage: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>Storage: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>Storage: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>Storage: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>Storage: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>Storage: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

### Storage Collection node

A node to represent a collection of Storage.

### Storage Collection node lifecycle

The following section describes the scenarios in which a Storage Collection node is created, updated or removed.

#### Creation/update

Storage Collection nodes are created and updated in exactly the same way as Host nodes.
Removal

A Storage Collection node is removed in one of the following ways:

- Manually, by selecting the necessary Storage Collection node, and picking Destroy from the Actions list.
- Explicitly by the pattern that created it. See Removal in the Pattern Language guide.
- If the evidence for a Storage Collection is not seen on a scan then the node is removed by reasoning. This is an Authoritative Removal type, see Authoritative Removal.

Storage Collection node attributes

The attributes and relationships on a Storage Collection node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of Storage Collection.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Vendor.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name that the StorageCollection is known by.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the StorageCollection. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>url string</td>
<td>URL for information about the StorageCollection. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>business_continuity_critical boolean</td>
<td>If true, the StorageCollection is critical to operation of the business. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>third_party_support boolean</td>
<td>If true, the StorageCollection is supported by a third party. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>synonyms string</td>
<td>Other names by which this StorageCollection is known. Legacy attribute not currently used. Can be used by patterns if desired.</td>
</tr>
</tbody>
</table>

Storage Collection node relationships

The relationships on a Storage Collection node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used by Host</td>
<td>StorageCollection: Storage: Storage: Host: Host</td>
<td>Host with Storage.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>Maintaining</td>
<td>StorageContainer: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this Storage.</td>
</tr>
<tr>
<td>Pattern</td>
<td>StorageContainer: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>StorageContainer: AttachmentContainer: Attachment: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>StorageContainer: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>StorageContainer: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>StorageContainer: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>StorageContainer: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person responsible for the support of this element.</td>
</tr>
<tr>
<td>Business Owner</td>
<td>StorageContainer: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>StorageContainer: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

### Generic Element node

The GenericElement's main role is for rapid prototyping of TPL patterns where the default BMC Atrium Discovery taxonomy does not already contain a more suitable node kind.

As the GenericElement is available by default, TPL can create it as required without needing to extend the taxonomy. For this reason there are few defined relationships; it is impossible to anticipate what will be needed.

It is recommended that the `type` attribute is set, so that the nodekind is logically sub-typed for multiple concurrent uses.

Once prototyping is finished it is expected that a more specific nodekind and associated taxonomy extension is considered.

#### Generic Element node lifecycle

The following section describes the scenarios in which a Generic Element node is created, updated or removed.

**Creation/update**

This is under the full control of patterns and as a result there is no default Generic Element node behavior.

The generated key for a Generic Element node is entirely dependent on the pattern that creates the Generic Element node.
Removal

A Generic Element node can be destroyed in one of the following ways:

- Manually, by selecting the necessary Detail node, and picking **Destroy** from the **Actions** list.
- Explicitly by the pattern that created it. See **Removal** in the **Pattern Language guide**
- When the last related Inferred node that is linked to the Generic Element node is removed, so will the Generic Element node. This is a Cascade Removal type, see **Cascade Removal**.

**Generic Element node attributes**

The attributes and relationships on a Generic Element node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of the detail.</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of detail.</td>
</tr>
<tr>
<td>Trigger Count</td>
<td>count string</td>
<td>Trigger count.</td>
</tr>
</tbody>
</table>

**Generic Element node relationships**

The relationships on a Generic Element node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Container</td>
<td>GenericElement: Contained: Containment: Container: GenericElement</td>
<td>Element this one is contained within.</td>
</tr>
<tr>
<td>Contained Elements</td>
<td>GenericElement: Container: Containment: Contained: GenericElement</td>
<td>Elements contained within this Element.</td>
</tr>
<tr>
<td>Elements This Depends On</td>
<td>GenericElement: Dependant: Dependency: DependedUpon: GenericElement</td>
<td>Element this one depends upon.</td>
</tr>
<tr>
<td>Elements That Depend On This</td>
<td>GenericElement: DependedUpon: Dependency: Dependant: GenericElement</td>
<td>Elements depending upon this Element.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>GenericElement: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern maintaining this Element.</td>
</tr>
</tbody>
</table>
Runtime Environment node

A Runtime Environment node represents an instance of a runtime environment on a Host, for example a Java Virtual Machine or a .NET runtime. Where possible, versions of Runtime Environments are retrieved and stored. They are created, maintained and destroyed by patterns. Runtime Environment nodes always have a relationship to the Pattern node corresponding to their maintaining pattern.

Why a Runtime Environment and not an SI?

Software Instances represent pieces of software running on a host. A Runtime Environment represents something supporting running software. For example, Weblogic is represented by an SI, and Java by a Runtime Environment node. However, it is still important to keep track of the supporting Runtime Environments and their versions as some might require updates.

Runtime Environment lifecycle

The lifecycle of a Runtime Environment node depends upon the details of the pattern that maintains it. This is determined by a combination of the pattern trigger condition, whether the pattern specifies an explicit key for the Runtime Environment and the logic contained in the pattern itself. In all cases, if a pattern is deleted, the Runtime Environment nodes it is maintaining are immediately destroyed (as are all other nodes it might be maintaining).

Creation/update

If a pattern triggers on a Directly Discovered Data node, such as a Discovered Process node, it might choose whether to specify keys for the Runtime Environment nodes it creates and maintains. If a key is specified then the decision whether to create a new Runtime Environment node or to update an existing one depends on the key. If a Runtime Environment node with the specified key exists, that node is updated, even if the node was previously maintained by a different pattern. In this case, the pattern takes over as the maintainer of the Runtime Environment. If a node with the specified key does not exist, a new Runtime Environment node is created. In both cases, the Runtime Environment node is linked to the pattern with a maintainer relationship.

If a key for the Runtime Environment node is not specified by the pattern, the system creates or updates a group Runtime Environment with an automatically generated key. The key is based upon the key of the Host upon which the Runtime Environment is running, the specified type of the Runtime Environment and, optionally, a key group that can be used to separate the nodes into a number of groups. The count attribute is set to the number of instances in the group identified in the collection of Directly Discovered Data. Each time the host is scanned, the count attribute is changed to represent the number of instances seen in that scan.
Removal

A Runtime Environment node can be destroyed either manually or automatically.

To remove a Runtime Environment node manually, find the necessary Runtime Environment, select it in the list and pick **Destroy** from the **Actions** list.

Automatic removal happens according to the following scenario.

The `age_count` attribute of the Runtime Environment node contains information about when the Runtime Environment node was last confirmed by its maintaining pattern. If the `age_count` is positive, it represents the number of consecutive scans of the Host node in which the Runtime Environment was confirmed. If the `age_count` is negative, it represents the number of consecutive scans in which the Runtime Environment node was not confirmed. The `last_update_success` and `last_update_failure` attributes contain the date and time at which the Runtime Environment node was last confirmed, and not confirmed, respectively.

The default aging strategy only applies to Runtime Environment nodes created from patterns triggering on the following node kinds and maintaining the Runtime Environments:

- DiscoveredProcess
- DiscoveredService
- DiscoveredSoftware

If the Runtime Environment node is triggered on anything else, for example, a discovered file, then aging must be implemented in the pattern using a removal block.

If the pattern does not have a removal block, Runtime Environment nodes are removed using an aging strategy based on the `age_count` and `last_update_success` attributes. The default aging parameters are the same as for a Software Instance node, that is, if a Runtime Environment node has not been seen for at least 7 scans, over a period of at least 10 days, it is destroyed. The parameters can be changed in the options, see Configuring model maintenance settings for more information.

If the pattern maintaining a node does have a removal block, the block can override the default aging scheme to destroy its nodes either earlier or later than normal. For TKU patterns, refer to the documentation accompanying each pattern for details of special removal behavior.

Regardless of the presence or absence of a removal block in the pattern, if the Host corresponding to a DDD-triggered Runtime Environment node is destroyed, the Runtime Environment node is immediately destroyed (see How nodes get removed).

Runtime Environment nodes currently represent Java or .NET. The different technologies and discovery techniques lead to the following differences in their removal:

- Java is grouped by command line; it has aging behavior like any grouped Software Instance.
• .NET is based on package information, so it has authoritative removal.

⚠️ It is possible for other triggers to be used. If any other trigger is used, BMC Atrium Discovery has no automatic removal behavior. Patterns must be used to explicitly destroy any Runtime Environment nodes created as a result of other triggers.

Runtime Environment node attributes

The attributes of a Runtime Environment node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of the Runtime Environment.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name of the Runtime Environment.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of the Runtime Environment.</td>
</tr>
<tr>
<td>Instance Count</td>
<td>count int</td>
<td>Number of instances grouped together.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Publisher</td>
<td>publisher string</td>
<td>The publisher of the Runtime Environment. Only populated in cases that a Pattern identifies products from more than one publisher. This information is normally found in the Pattern node's publishers attribute.</td>
</tr>
<tr>
<td>Product Name</td>
<td>product string</td>
<td>The product name. Only populated in cases that a Pattern identifies more than one product. This information is normally found in the Pattern node's products attribute.</td>
</tr>
<tr>
<td>Full Version</td>
<td>version string</td>
<td>Full-resolution version.</td>
</tr>
<tr>
<td>Product Version</td>
<td>product_version string</td>
<td>Version publicised by the vendor.</td>
</tr>
<tr>
<td>Release</td>
<td>release string</td>
<td>Release number.</td>
</tr>
<tr>
<td>Service Pack</td>
<td>service_pack string</td>
<td>Service pack.</td>
</tr>
<tr>
<td>Build</td>
<td>build string</td>
<td>Build number.</td>
</tr>
<tr>
<td>Patch</td>
<td>patch string</td>
<td>Patch level.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>Revision</td>
<td>Revision.</td>
<td></td>
</tr>
<tr>
<td>age_count int</td>
<td>The number of consecutive successful (positive) or failed (negative) times the Runtime Environment has been seen during host scans.</td>
<td></td>
</tr>
<tr>
<td>last_update_success date</td>
<td>The time at which a scan was last successfully associated with this Runtime Environment.</td>
<td></td>
</tr>
<tr>
<td>last_update_failure date</td>
<td>The time at which a scan associated with this Runtime Environment failed.</td>
<td></td>
</tr>
<tr>
<td>_explicit_removal_string</td>
<td>Pattern that has taken over removal in place of standard removal.</td>
<td></td>
</tr>
</tbody>
</table>

**Runtime Environment node relationships**

The attributes and relationships of a Runtime Environment node are described in the tables below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>RuntimeEnvironment: RunningSoftware: HostedSoftware: Host: Host</td>
<td>Host on which this Runtime Environment is present.</td>
</tr>
<tr>
<td>MFFPart</td>
<td>RuntimeEnvironment: RunningSoftware: HostedSoftware: Host: MFFPart</td>
<td>MFFPart on which this Runtime Environment is present.</td>
</tr>
<tr>
<td>Maintaining Pattern</td>
<td>RuntimeEnvironment: Element: Maintainer: Pattern: Pattern</td>
<td>Pattern that is maintaining this Runtime Environment.</td>
</tr>
<tr>
<td>Depended Upon By Software Instances</td>
<td>RuntimeEnvironment: DependedUpon: Dependency: Dependant: SoftwareInstance</td>
<td>Software Instances that depend upon this Runtime Environment.</td>
</tr>
<tr>
<td>Details</td>
<td>RuntimeEnvironment: ElementWithDetail: Detail: Detail</td>
<td>Details of this Runtime Environment.</td>
</tr>
<tr>
<td>Primary processes</td>
<td>RuntimeEnvironment: InferredElement: Inference: Primary: DiscoveredProcess</td>
<td>Discovered process from which the existence of this Runtime Environment was inferred.</td>
</tr>
<tr>
<td>Contributor processes</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredProcess</td>
<td>Discovered process from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated processes</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredProcess</td>
<td>Discovered process related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Associated services</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredService</td>
<td>Discovered service related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: Package</td>
<td>Package from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated packages</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: Package</td>
<td>Package related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: Host</td>
<td>Host from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: Host</td>
<td>Host related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Primary: DiscoveredFile</td>
<td>Discovered file from which the existence of this Runtime Environment was inferred.</td>
</tr>
<tr>
<td>Associated files</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredFile</td>
<td>Discovered file related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Primary software</td>
<td>RuntimeEnvironment: InferredElement: Inference: Primary: DiscoveredSoftware</td>
<td>Discovered software from which the existence of this Runtime Environment was inferred.</td>
</tr>
<tr>
<td>Contributor software</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredSoftware</td>
<td>Discovered software from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated discovered software</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredSoftware</td>
<td>Discovered software related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredCommandResult</td>
<td>Discovered command result from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated command results</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredCommandResult</td>
<td>Discovered command result related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated registry values</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredRegistryValue</td>
<td>Discovered Windows Registry value related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredWMI</td>
<td>Discovered WMI query result from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated WMI values</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredWMI</td>
<td>Discovered WMI query result related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Associated WBEM instances</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredWBEMInstance</td>
<td>Discovered WBEM instance related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Associated WBEM query results</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM result related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: DiscoveredWBEMQueryResult</td>
<td>Discovered WBEM result from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Associated WBEM associators results</td>
<td>RuntimeEnvironment: InferredElement: Inference: Associate: DiscoveredWBEMAssociatorsResult</td>
<td>Discovered WBEM association related in some way to this Runtime Environment.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: Pattern</td>
<td>Pattern from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Primary: MFFPart</td>
<td>MFFPart from which the existence of this Runtime Environment was inferred.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>RuntimeEnvironment: InferredElement: Inference: Contributor: MFFPart</td>
<td>MFFPart from which one or more attributes of this Runtime Environment were inferred.</td>
</tr>
</tbody>
</table>

**Storage nodes**

The following nodes are used to model storage systems:

- Storage Device node
- Storage System node
- Storage Pool node
- Storage System Group node
- Storage Volume node
- Storage Connection node
- Storage Processor node
- Disk Drive node

**Storage Device node**

The model of a storage entity consists of a central Storage System node, and associated Storage pools, Storage volumes, Storage processors, Front end FC ports, and Storage connections. Directly discovered storage entities also have a Storage Device node. The Storage Device node can be regarded in much the same way as the SI representing the storage management software. They are the trigger for the discovery of the storage entity rather than an integral component.
Creation/update

A Storage Device node is created or updated when a storage device is discovered either through discovery of the WBEM/SMI-S provider in the storage chassis, or a NetApp device is discovered using SNMP.

Removal

The removal of a Storage Device node is governed by aging in the same way as Host nodes.

When a Storage Device node is removed, all of the nodes it contains are removed. This is Containment Removal, see Containment Removal.

Storage Device node attributes

The attributes of a Storage Device node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>model string</td>
<td>Model of device.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Vendor of device.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>serial string</td>
<td>Serial.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of device.</td>
</tr>
<tr>
<td>World Wide Node Name</td>
<td>wwnn string</td>
<td>World Wide Node Name.</td>
</tr>
<tr>
<td>System Name</td>
<td>sysname string</td>
<td>System name.</td>
</tr>
<tr>
<td>System Object ID</td>
<td>sysobjectid string</td>
<td>System object id.</td>
</tr>
<tr>
<td>Capability IDs</td>
<td>_capability_ids _list:int</td>
<td>Capability type ids of device.</td>
</tr>
<tr>
<td>Capability</td>
<td>capability_types list:string</td>
<td>Capability types of device.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of device.</td>
</tr>
<tr>
<td>Firmware Version</td>
<td>firmware_version string</td>
<td>Firmware version.</td>
</tr>
<tr>
<td>Testing Status</td>
<td>testing_status string</td>
<td>Device testing status.</td>
</tr>
<tr>
<td>OS Type</td>
<td>os_type string</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>OS Class</td>
<td>os_class string</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>OS Version</td>
<td>os_version string</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>OS Vendor</td>
<td>os_vendor string</td>
<td>Operating system vendor.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__all_ip_addrs list:string</td>
<td>Internal attribute to aid searching Devices by IP address.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__all_dns_names list:string</td>
<td>Internal attribute to aid searching Devices by name.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>age_count int</td>
<td>The number of consecutive successful (positive) or failed (negative) accesses, from any endpoint.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_success date</td>
<td>The time at which a scan was last successfully associated with this device.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>last_update_failure date</td>
<td>The time at which a scan associated with this device failed.</td>
</tr>
<tr>
<td>Last successful CMDB sync</td>
<td>last_cmdb_sync_success date</td>
<td>The time at which this device was last successfully synchronized into the CMDB.</td>
</tr>
<tr>
<td>Last failed CMDB sync</td>
<td>last_cmdb_sync_failure date</td>
<td>The time at which an attempt to synchronize this device into the CMDB failed.</td>
</tr>
<tr>
<td>CMDB sync duration</td>
<td>last_cmdb_sync_duration float</td>
<td>The time in seconds spent performing the last CMDB synchronization of this device.</td>
</tr>
<tr>
<td>CMDB CI count</td>
<td>last_cmdb_sync_ci_count int</td>
<td>The number of CIs corresponding to this device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>CMDB Relationship count</td>
<td>last_cmdb_sync_rel_count int</td>
<td>The number of relationships between CIs corresponding to this device at the last CMDB synchronization.</td>
</tr>
<tr>
<td>Cluster</td>
<td>cluster : boolean</td>
<td>Flag to indicate if this is a Cluster.</td>
</tr>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Primary name.</td>
</tr>
<tr>
<td>Short Name</td>
<td>short_name string</td>
<td>Short name.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description of the element.</td>
</tr>
<tr>
<td>URL</td>
<td>url string</td>
<td>URL for information about the element.</td>
</tr>
<tr>
<td>If true, element is critical to operation of the business.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### UI Name and Type

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Continuity Critical</td>
<td>business_continuity_critical boolean</td>
<td></td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>third_party_support boolean</td>
<td>True if the element is supported by a third party.</td>
</tr>
<tr>
<td>Synonyms</td>
<td>synonyms list:string</td>
<td>Other names for the element.</td>
</tr>
</tbody>
</table>

### Storage Device node relationships

The relationships of a Storage Device node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Access</td>
<td>StorageDevice: InferredElement: Inference: Associate: DiscoveryAccess</td>
<td>DiscoveryAccess for this device.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>StorageDevice: InferredElement: Inference: Primary: DeviceInfo</td>
<td>DeviceInfo of this device.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: NetworkDevice</td>
<td>Previous Network Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: Host</td>
<td>Previous Host Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>StorageDevice: Previous: EndpointIdentity: Next: Host</td>
<td>Next Host Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: Mainframe</td>
<td>Previous Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>StorageDevice: Previous: EndpointIdentity: Next: Mainframe</td>
<td>Next Mainframe Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: Printer</td>
<td>Previous Printer Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>StorageDevice: Previous: EndpointIdentity: Next: Printer</td>
<td>Next Printer Identity.</td>
</tr>
</tbody>
</table>

Previous SNMP Managed Device Identity.
<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: SNMPManagedDevice</td>
<td></td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>StorageDevice: Previous: EndpointIdentity: Next: SNMPManagedDevice</td>
<td>Next SNMP Managed Device Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: ManagementController</td>
<td>Previous Management Controller Identity.</td>
</tr>
<tr>
<td>Endpoint Used To Be Identified As</td>
<td>StorageDevice: Next: EndpointIdentity: Previous: StorageDevice</td>
<td>Previous Storage Device Identity.</td>
</tr>
<tr>
<td>Endpoint Now Identified As</td>
<td>StorageDevice: Previous: EndpointIdentity: Next: StorageDevice</td>
<td>Next Storage Device Identity.</td>
</tr>
<tr>
<td>Chosen Endpoint</td>
<td>StorageDevice: Device: ChosenEndpoint: Endpoint</td>
<td>Endpoint used to discover this device.</td>
</tr>
<tr>
<td>Network Interfaces</td>
<td>StorageDevice: DeviceWithInterface: DeviceInterface: InterfaceOfDevice: NetworkInterface</td>
<td>Network interfaces of this device.</td>
</tr>
<tr>
<td>IPv4 Addresses</td>
<td>StorageDevice: DeviceWithAddress: DeviceAddress: IPv4Address: IPAddress</td>
<td>IPv4 addresses of this device.</td>
</tr>
<tr>
<td>IPv6 Addresses</td>
<td>StorageDevice: DeviceWithAddress: DeviceAddress: IPv6Address: IPAddress</td>
<td>IPv6 addresses of this device.</td>
</tr>
<tr>
<td>Storage System</td>
<td>StorageDevice: Manager: Management: ManagedElement: StorageSystem</td>
<td>Storage system on this device.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>StorageDevice: AttachmentContainer: Attachment: Attachment</td>
<td>A file attached to this element.</td>
</tr>
<tr>
<td>Status</td>
<td>StorageDevice: ElementInCategory: ElementCategory: Category: LifecycleStatus</td>
<td>The lifecycle status of this element.</td>
</tr>
<tr>
<td>Recovery Time</td>
<td>StorageDevice: ElementInCategory: ElementCategory: Category: RecoveryTime</td>
<td>The recovery time specified for this element.</td>
</tr>
<tr>
<td>Family</td>
<td>StorageDevice: ElementInCategory: ElementCategory: Category: Family</td>
<td>Family of this element.</td>
</tr>
<tr>
<td>Organizational Unit</td>
<td>StorageDevice: OwnedItem: Ownership: Owner: OrganisationalUnit</td>
<td>The Organizational Unit that this element belongs to.</td>
</tr>
<tr>
<td>Location</td>
<td>StorageDevice: ElementInLocation: Location: Location: Location</td>
<td>Location of this element.</td>
</tr>
<tr>
<td>Support Manager</td>
<td>StorageDevice: OwnedItem: Ownership: SupportOwner: Person</td>
<td>The person or owner responsible for the support of this element.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Business Owner</td>
<td>StorageDevice: OwnedItem: Ownership: BusinessOwner: Person</td>
<td>The person or owner responsible for this element from a business perspective.</td>
</tr>
<tr>
<td>IT Owner</td>
<td>StorageDevice: OwnedItem: Ownership: ITOwner: Person</td>
<td>The person or owner responsible for this element from an IT perspective.</td>
</tr>
</tbody>
</table>

**Storage System node**

A Storage System node represents a single storage array chassis. It is the container node for the other nodes modeling the structure of storage entities.

**Creation/update**

A Storage System node is created or updated when a storage entity is discovered by one of the following:

- WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
- Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the onboard controller of the storage array chassis
- Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the onboard controller of the storage array chassis

**Removal**

The removal of a Storage System node is governed by aging in the same way as Host nodes. However, in the case of the Storage System, it is the relationships between the Storage System and its manager that are aged. The manager is either:

- The SI representing the management software
- The Storage Device representing the onboard controller of the storage array chassis

**Case 1 - Single manager**

When the manager is scanned and the Storage System is no longer discovered, the management relationship begins to age. If it is not discovered in ten days and seven scans (the default), the relationship is removed. If the Storage System has no relationships to a manager, then it is removed.

**Case 2 - Multiple managers**

Where a Storage System has multiple managers, SIs, Storage Devices, or both, each management relationship ages in the same way as the single manager case. If the Storage System is no longer discovered through a manager, then the relationship between that manager and the Storage System begins to age, and after ten days and seven scans, the relationship is
removed. If during this time, the Storage system has been successfully discovered at each scan through a different manager, then the relationship between that manager and the Storage System does not age. Consequently, the Storage System still has a relationship with a manager, and is not removed. When all relationships have been removed, the Storage System is removed too.

When a StorageSystem is removed, all of the nodes it contains are removed. This is Containment Removal, see Containment Removal.

**Storage System node attributes**

The attributes of a Storage System node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>type string</td>
<td>Storage System type.</td>
</tr>
<tr>
<td>Storage Access Mechanisms</td>
<td>access_types : list string</td>
<td>Storage access mechanisms.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Device model.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Device manufacturer.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>serial string</td>
<td>Serial number.</td>
</tr>
<tr>
<td>World Wide Node Name</td>
<td>wwnn string</td>
<td>World Wide Node Name.</td>
</tr>
<tr>
<td>Software Version</td>
<td>sw_version string</td>
<td>Software version.</td>
</tr>
<tr>
<td>Total Disk Space</td>
<td>total_disk_space int</td>
<td>Total disk space.</td>
</tr>
<tr>
<td>Total Capacity</td>
<td>total_capacity int</td>
<td>Total capacity.</td>
</tr>
<tr>
<td>Total Consumed Capacity</td>
<td>total_consumed_capacity int</td>
<td>Total consumed capacity.</td>
</tr>
<tr>
<td>Total Subscribed Capacity</td>
<td>total_subscribed_capacity int</td>
<td>Total subscribed capacity.</td>
</tr>
<tr>
<td>Allocated Capacity</td>
<td>allocated_capacity int</td>
<td>Allocated capacity.</td>
</tr>
<tr>
<td>Available Capacity</td>
<td>available_capacity int</td>
<td>Available capacity.</td>
</tr>
<tr>
<td>Backend Storage Capacity</td>
<td>backend_storage_capacity int</td>
<td>Backend storage capacity.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__all_ip_addrs list:string</td>
<td>Internal attribute to aid searching Devices by IP address.</td>
</tr>
</tbody>
</table>
**UI Name**

<table>
<thead>
<tr>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI</td>
<td>Internal attribute to aid searching Devices by name.</td>
</tr>
<tr>
<td>__all_dns_names list:string</td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>key string</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Primary name.</td>
</tr>
<tr>
<td>name string</td>
<td></td>
</tr>
<tr>
<td>Short Name</td>
<td>Short name.</td>
</tr>
<tr>
<td>short_name string</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td>Description of the element.</td>
</tr>
<tr>
<td>description string</td>
<td></td>
</tr>
<tr>
<td>URL</td>
<td>URL for information about the element.</td>
</tr>
<tr>
<td>url string</td>
<td></td>
</tr>
<tr>
<td>Business Continuity Critical</td>
<td>If true, element is critical to operation of the business.</td>
</tr>
<tr>
<td>business_continuity_critical boolean</td>
<td></td>
</tr>
<tr>
<td>Supported by 3rd Party</td>
<td>True if the element is supported by a third party.</td>
</tr>
<tr>
<td>third_party_support boolean</td>
<td></td>
</tr>
<tr>
<td>Synonyms</td>
<td>Other names for the element.</td>
</tr>
<tr>
<td>synonyms list:string</td>
<td></td>
</tr>
</tbody>
</table>

**Storage System node relationships**

The relationships of a Storage System node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk Drives</td>
<td>StorageSystem: Container: Containment: ContainedDrive: DiskDrive</td>
<td>Disk drives for this StorageSystem.</td>
</tr>
<tr>
<td>Storage Device</td>
<td>StorageSystem: ManagedElement: Management: Manager: StorageDevice</td>
<td>Storage device of this Storage System.</td>
</tr>
<tr>
<td>Managed By</td>
<td>StorageSystem: ManagedElement: Management: Manager: SoftwareInstance</td>
<td>Software Instances that manage this Storage System.</td>
</tr>
<tr>
<td>Managed By</td>
<td>StorageSystem: ManagedElement: Management: Manager: NetworkDevice</td>
<td>NetworkDevices that manage this Storage System.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The group that this Storage system is part of.</td>
</tr>
</tbody>
</table>
### UI name | Relationship | Description
--- | --- | ---
Failover Group | StorageSystem: ContainedInstance: Containment: Container: StorageSystemGroup |  
Not displayed in UI | StorageSystem: AttachmentContainer: Attachment: Attachment | A file attached to this element.
Status | StorageSystem: ElementInCategory: ElementCategory: Category: LifecycleStatus | The lifecycle status of this element.
Recovery Time | StorageSystem: ElementInCategory: ElementCategory: Category: RecoveryTime | The recovery time specified for this element.
Family | StorageSystem: ElementInCategory: ElementCategory: Category: Family | Family of this element.
Organizational Unit | StorageSystem: OwnedItem: Ownership: Owner: OrganisationalUnit | The Organizational Unit that this element belongs to.
Location | StorageSystem: ElementInLocation: Location: Location | Location of this element.
Support Manager | StorageSystem: OwnedItem: Ownership: SupportOwner: Person | The person or owner responsible for the support of this element.
Business Owner | StorageSystem: OwnedItem: Ownership: BusinessOwner: Person | The person or owner responsible for this element from a business perspective.
IT Owner | StorageSystem: OwnedItem: Ownership: ITOwner: Person | The person or owner responsible for this element from an IT perspective.

### Storage Pool node

A Storage Pool node represents a block of storage from which storage volumes and other storage pools can be allocated. A storage system may expose one or more storage pools. Where there is a hierarchy of pools (Primordial -> Concrete), this is modeled using a Hierarchy relationship.

### Creation/update

A Storage Pool node is created or updated when a storage pool is discovered as a result of:

- WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
- Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the chassis
- Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the chassis

### Removal

A Storage Pool node is removed when its parent Storage System node is removed. This is Containment Removal.

### Storage Pool node attributes

The attributes of a Storage Pool node are described in the table below.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI key string</td>
<td></td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Pool ID pool_id string</td>
<td></td>
<td>Id of the pool.</td>
</tr>
<tr>
<td>Type pool_type string</td>
<td></td>
<td>Type of the pool.</td>
</tr>
<tr>
<td>Name name string</td>
<td></td>
<td>Name of the pool.</td>
</tr>
<tr>
<td>RAID Level raid_level string</td>
<td></td>
<td>RAID level.</td>
</tr>
<tr>
<td>Capacity capacity int</td>
<td></td>
<td>Capacity in bytes.</td>
</tr>
<tr>
<td>Consumed Capacity consumed_capacity int</td>
<td></td>
<td>Amount of capacity that has actually been consumed by clients.</td>
</tr>
<tr>
<td>Subscribed Capacity subscribed_capacity int</td>
<td></td>
<td>Amount of capacity that has been assigned to clients.</td>
</tr>
<tr>
<td>Oversubscription Warning oversubscription_warning boolean</td>
<td></td>
<td>Indicates that the pool is over-subscribed and running low on capacity.</td>
</tr>
<tr>
<td>Remaining Capacity remaining_capacity int</td>
<td></td>
<td>Remaining capacity in bytes.</td>
</tr>
<tr>
<td>Total Raw Capacity total_raw_capacity int</td>
<td></td>
<td>Total raw capacity in bytes.</td>
</tr>
<tr>
<td>Remaining Raw Capacity remaining_raw_capacity int</td>
<td></td>
<td>Remaining raw capacity in bytes.</td>
</tr>
<tr>
<td>Virtual Capacity virtual_capacity int</td>
<td></td>
<td>Virtual capacity in bytes.</td>
</tr>
<tr>
<td>Real Capacity real_capacity int</td>
<td></td>
<td>Real capacity in bytes.</td>
</tr>
<tr>
<td>Virtual virtual : boolean</td>
<td></td>
<td>Flag to indicate if this is virtual.</td>
</tr>
</tbody>
</table>

**Storage Pool node relationships**

The relationships of a Storage Pool node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>
Storage System Group node

A Storage System Group node represents a group of connected storage systems. Currently this is only for NetApp failover pairs.

Creation/update

A Storage System Group node is created or updated when a Storage System node is created which is one member of a failover pair. When the second member is discovered, it is linked to the corresponding Storage System Group node.

Removal

A Storage System Group node is removed when either member of the failover group is scanned and is no longer in a pair relationship. Also, if both Storage System nodes age out, the Storage System Group node is removed.

Storage System Group node attributes

The attributes of a Storage System Group node are described in the table below:

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Type</td>
<td>type string</td>
<td>Type of storage system group.</td>
</tr>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name of storage system group.</td>
</tr>
</tbody>
</table>

Storage System Group node relationships

The relationships of a Storage System Group node are described in the table below:

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Systems</td>
<td>StorageSystemGroup: Container: Containment:</td>
<td>Storage system instances that are part of this group.</td>
</tr>
<tr>
<td></td>
<td>ContainedInstance: StorageSystem</td>
<td></td>
</tr>
</tbody>
</table>

Storage Volume node

A Storage Volume node represents a block of storage which a client can access.
Creation/update

A Storage Volume node is created or updated when a storage entity is discovered by one of the following:

- WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
- Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the chassis
- Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the chassis

Removal

A Storage Volume node is removed when its parent Storage System node is removed. This is Containment Removal.

Storage Volume node attributes

The attributes of a Storage Volume node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI key string</td>
<td>Key of the volume.</td>
</tr>
<tr>
<td>Volume ID volume_id string</td>
<td>Identity of the volume.</td>
</tr>
<tr>
<td>Type type string</td>
<td>Type of the volume.</td>
</tr>
<tr>
<td>NAA ID naa_id string</td>
<td>NAA ID (WWN).</td>
</tr>
<tr>
<td>Name name string</td>
<td>Name of the volume.</td>
</tr>
<tr>
<td>Block Size block_size int</td>
<td>Block size.</td>
</tr>
<tr>
<td>Number of Blocks num_blocks int</td>
<td>Number of blocks.</td>
</tr>
<tr>
<td>Consumable Blocks consumable_blocks int</td>
<td>Consumable blocks.</td>
</tr>
<tr>
<td>Consumed From Pool consumed_capacity int</td>
<td>Amount of capacity that has been consumed in the parent pool. For thin volumes, this value will typically be less than the visible capacity; for traditional volumes this value equals, or is greater than, the visible capacity. Indicates a thinly provisioned volume.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Thinly Provisioned</td>
<td>thinly_provisioned boolean</td>
</tr>
<tr>
<td>Mapped</td>
<td>mapped boolean</td>
</tr>
</tbody>
</table>

**Storage Volume node relationships**

The relationships of a Storage Volume node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Volume Providers</td>
<td>StorageVolume: Consumer: StorageUse: Provider: StorageVolume</td>
<td>Storage volumes providing this Volume.</td>
</tr>
<tr>
<td>Storage Volume Consumers</td>
<td>StorageVolume: Provider: StorageUse: Consumer: StorageVolume</td>
<td>Storage volumes consuming this Volume.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>StorageVolume: Provider: StorageUse: Consumer: DiskDrive</td>
<td>Virtual disks consuming this Volume.</td>
</tr>
<tr>
<td>Storage Consumers</td>
<td>StorageVolume: Container: Containment: ContainedConnection: StorageConnection</td>
<td>Consumers of this volume.</td>
</tr>
<tr>
<td>Front End FC Ports</td>
<td>StorageVolume: ExposedElement: ExposedView: FrontEndPort: FibreChannelPort</td>
<td>Front end FC ports for this Volume.</td>
</tr>
</tbody>
</table>

**Storage Connection node**

A Storage Connection node represents a connection between a storage client (a host using the storage) and a storage volume. The same connection may be linked to multiple volumes representing the client's visibility of multiple storage volumes.

**Creation/update**

A Storage Connection node is created or updated when a storage entity is discovered by one of the following:

- WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
• Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the chassis
• Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the chassis

Removal
A StorageConnection node is removed when its parent Storage System node is removed. This is Containment Removal.

Storage Connection node attributes
The attributes of a Storage Connection node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI</td>
<td>key string</td>
<td>Key of the connection details.</td>
</tr>
<tr>
<td>Connection Type</td>
<td>type string</td>
<td>Type of connection.</td>
</tr>
<tr>
<td>Connection Value</td>
<td>value string</td>
<td>Connection value.</td>
</tr>
<tr>
<td>Connection Name</td>
<td>name string</td>
<td>Useful name of the connection.</td>
</tr>
</tbody>
</table>

Storage Connection node relationships
The relationships of a Storage Connection node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage Volumes</td>
<td>StorageConnection: ContainedConnection: Containment: Container: StorageVolume</td>
<td>Storage volumes for this connection.</td>
</tr>
<tr>
<td>Consumer Fibre Channel Port</td>
<td>StorageConnection: StorageProducer: SANStorage: StorageConsumer: FibreChannelPort</td>
<td>FC port connected to this connection.</td>
</tr>
</tbody>
</table>

Storage Processor node
Represents a processor in a storage system.

Creation/update
A Storage Processor node is created or updated when a storage entity is discovered by one of the following:

• WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
• Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the chassis
• Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the chassis

Removal
A Storage Processor node is removed when its parent Storage System node is removed. This is Containment Removal.

Storage Processor node attributes
The attributes of a Storage Processor node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Unique key.</td>
</tr>
<tr>
<td>Processor Identifier</td>
<td>processor_id string</td>
<td>Processor identifier.</td>
</tr>
<tr>
<td>Vendor</td>
<td>vendor string</td>
<td>Hardware vendor.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Model name.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Model description.</td>
</tr>
<tr>
<td>Serial</td>
<td>serial string</td>
<td>Serial number.</td>
</tr>
<tr>
<td>Firmware</td>
<td>firmware string</td>
<td>Firmware details.</td>
</tr>
<tr>
<td>Memory</td>
<td>memory int</td>
<td>Memory size in bytes.</td>
</tr>
<tr>
<td>PROM revision</td>
<td>prom_revision string</td>
<td>PROM revision.</td>
</tr>
</tbody>
</table>

Storage Processor node relationships
The relationships of a Storage Processor node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage System</td>
<td>StorageProcessor: ContainedProcessor: Containment:</td>
<td>StorageSystem for this processor.</td>
</tr>
<tr>
<td></td>
<td>Container: StorageSystem</td>
<td></td>
</tr>
<tr>
<td>Front End FC Ports</td>
<td>StorageProcessor: Contained: Contained: FibreChannelPort</td>
<td>Front End FC Ports for this processor.</td>
</tr>
<tr>
<td>Front End Ethernet Ports</td>
<td>StorageProcessor: Contained: Contained: NetworkInterface</td>
<td>Front End Ethernet Ports for this processor.</td>
</tr>
</tbody>
</table>
Disk Drive node
A Disk Drive node represents a physical disk drive in a storage system.

Creation/update
A Disk Drive node is created or updated when a storage entity is discovered by one of the following:

- WBEM queries to storage management software triggered by the creation or update of an SI representing storage management software
- Directly discovered storage using WBEM queries to the WBEM/SMI-S provider triggered by the creation or update of a Storage Device node representing the chassis
- Directly discovered storage using SNMP triggered by the creation or update of a Storage Device node representing the chassis

Removal
A Disk Drive node is removed when its parent Storage System node is removed. This is Containment Removal.

Disk Drive node attributes
The attributes of a Disk Drive node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key key string</td>
<td>Globally unique key.</td>
</tr>
<tr>
<td>Name name string</td>
<td>Name of the disk drive.</td>
</tr>
<tr>
<td>Type type string</td>
<td>Type of the disk drive.</td>
</tr>
<tr>
<td>Speed speed int</td>
<td>Speed of the disk drive in RPM.</td>
</tr>
<tr>
<td>Size size int</td>
<td>Size in bytes.</td>
</tr>
<tr>
<td>Vendor vendor string</td>
<td>Disk drive vendor.</td>
</tr>
<tr>
<td>Model model string</td>
<td>Disk drive model.</td>
</tr>
<tr>
<td>Serial Number serial string</td>
<td>Disk drive serial number.</td>
</tr>
<tr>
<td>Role role string</td>
<td>Role.</td>
</tr>
<tr>
<td></td>
<td>Current state of the disk drive.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Current State</td>
<td>current_state string</td>
</tr>
<tr>
<td>Virtual</td>
<td>virtual boolean</td>
</tr>
</tbody>
</table>

**Disk Drive node relationships**

The relationships of a Disk Drive node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage System</td>
<td>DiskDrive: ContainedDrive: Containment: Container: StorageSystem</td>
<td>StorageSystem for this disk.</td>
</tr>
<tr>
<td>Host</td>
<td>DiskDrive: ContainedDrive: Containment: Container: Host</td>
<td>Host for this disk.</td>
</tr>
<tr>
<td>Physical Disks</td>
<td>DiskDrive: Consumer: StorageUse: Provider: DiskDrive</td>
<td>Physical disk for this virtual disk.</td>
</tr>
<tr>
<td>Virtual Disks</td>
<td>DiskDrive: Provider: StorageUse: Consumer: DiskDrive</td>
<td>Virtual disks using this physical disk.</td>
</tr>
<tr>
<td>Physical Volumes</td>
<td>DiskDrive: Consumer: StorageUse: Provider: StorageVolume</td>
<td>Physical storage volume for this virtual disk.</td>
</tr>
</tbody>
</table>

**DDD nodes**

This section describes the BMC Atrium Discovery Directly Discovered Data (DDD) nodes. Directly discovered information is obtained directly from a target host via particular discovery techniques.

- **Introductory information** provides an introduction to the DDD nodes.
- **Discovery Run node** describes the Discovery Run node kind.
- **Discovery Access node** describes the Discovery Access node kind and how one is created each time a full discovery run is performed.
- **Device Info node** describes the Discovery Info node kind.
- **Host Info node** describes the Host Info node kind.
- **Other DDD nodes** describes the remaining Directly Discovered Data node kinds in brief.
- **Integration DDD nodes** describes the nodes related to Integration and SQL Discovery.

**Introductory information**

All Directly Discovered Data is stored, regardless of whether it is understood or not. Target discovery devices are accessed as part of a Discovery Run, corresponding to a range of devices that are scanned. When Reasoning starts accessing a target device, it creates a Discovery Access node to represent the access and relates it to the corresponding Discovery Run.
Access node contains information about the time of discovery and the discovery method and credentials used. Each time a device is rediscovered, a new Discovery Access node is created, and the discovery information is attached to it. If a target host cannot be accessed, a Discovery Access node is created to represent the failed access.

⚠️ **The Host node**

The Host node itself is not considered directly discovered information, since a level of inference is required to decide which Host a Discovery Access corresponds to. See Host node for more information.

Discovery Run nodes are created first because they contain all other DDD nodes. Discovery Access nodes are then created, followed by the remaining DDD nodes.

**Discovery Run node**

A Discovery Run is a scan of one or more Discovery endpoints, specified as an IP address or addresses or ranges which are scanned as an entity. These ranges might be being scanned locally on from the appliance, or be the result of appliance consolidation.

For each Discovery Run, a Discovery Run node is created which records information such as the user who started the run, the start and end time, and so on. A Discovery Run contains a number of Discovery Access nodes. In turn, Discovery Access nodes contain all other non-integration DDD nodes.

Additionally a Discovery Run node can also contain Provider Access nodes. A Provider Access node is created for a particular IP address as a result of a SQL Discovery access. In turn, Provider Access nodes contain all other Integration related DDD nodes.

**Discovery Run lifecycle**

The following section describes the scenarios in which a Discovery Run is created or destroyed. DDD nodes are never updated.

**Creation**

A Discovery Run node is created when a Discovery Run starts; when a range that has been configured to be scanned starts actually scanning or, if the appliance is a Consolidation Appliance, when a Scanning Appliance’s scan starts to be consolidated. The Discovery Run remains open until all of the endpoints (IP addresses) in the range have been scanned or consolidated.

**Removal**

Discovery Runs are removed when all of their contained Discovery Access have been destroyed through the Aging process. When the Discovery Run node is destroyed any linked Integration Access nodes and their associated DDD nodes are also destroyed.
# Discovery Run node attributes

The attributes and relationships of a Discovery Run node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>key string</td>
<td>Unique identity string for DiscoveryRun.</td>
</tr>
<tr>
<td>Label</td>
<td>label string</td>
<td>Label.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>Description.</td>
</tr>
<tr>
<td>User</td>
<td>user string</td>
<td>User who created the run.</td>
</tr>
<tr>
<td>Range Prefix</td>
<td>range_prefix string</td>
<td>Distinguish overlapping address spaces.</td>
</tr>
<tr>
<td>Valid Ranges</td>
<td>valid_ranges string</td>
<td>Valid IP address ranges in the run.</td>
</tr>
<tr>
<td>Total Number of IPs</td>
<td>total int</td>
<td>Number of addresses in run.</td>
</tr>
<tr>
<td>Number of IPs being Pre-scanned</td>
<td>pre_scanning int</td>
<td>Number of addresses being pre-scanned.</td>
</tr>
<tr>
<td>Number of IPs being Scanned</td>
<td>scanning int</td>
<td>Number of addresses being scanned.</td>
</tr>
<tr>
<td>Number of IPs waiting on Exclude Ranges</td>
<td>waiting int</td>
<td>Number of addresses waiting for end of exclude range.</td>
</tr>
<tr>
<td>Blocked Waiting on Exclude Ranges</td>
<td>blocked boolean</td>
<td>Blocked waiting for end of exclude range.</td>
</tr>
<tr>
<td>Number of IPs Scanned</td>
<td>done_int</td>
<td>Number of completed addresses.</td>
</tr>
<tr>
<td>Cancelled</td>
<td>cancelled string</td>
<td>If present reason why run was cancelled.</td>
</tr>
<tr>
<td>Consolidation Cancelled</td>
<td>consolidation_cancelled string</td>
<td>If present reason why consolidation run was cancelled.</td>
</tr>
<tr>
<td>Scan Type</td>
<td>scan_type string</td>
<td>Type of scan.</td>
</tr>
<tr>
<td>Scan Level</td>
<td>scan_level string</td>
<td>Scan level.</td>
</tr>
<tr>
<td>Start Time</td>
<td>starttime date</td>
<td>Date and time the run started.</td>
</tr>
<tr>
<td>End Time</td>
<td>endtime date</td>
<td>Date and time the run ended.</td>
</tr>
<tr>
<td>Discovery Start Time</td>
<td>discovery_starttime date</td>
<td>Date and time the discovery started.</td>
</tr>
<tr>
<td>Discovery End Time</td>
<td>discovery_endtime date</td>
<td>Date and time the discovery ended.</td>
</tr>
<tr>
<td>ECA Error</td>
<td>has_eca_error boolean</td>
<td>True if there have been any ECA Errors for this Discovery Run.</td>
</tr>
<tr>
<td>Hard Failure</td>
<td>hard_failure boolean</td>
<td>True if there have been hard failures for this Discovery Run.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__inprogress boolean</td>
<td>True if Discovery Run still executing.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__remote_inprogress boolean</td>
<td>True if Discovery Run still executing on the discovery appliance.</td>
</tr>
<tr>
<td>Number of IPs Consolidated</td>
<td>consolidation_done_count int</td>
<td>Number of completed consolidation endpoints.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__consolidation_source string</td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Consolidated From</td>
<td>_consolidation_source_name string</td>
<td>Name of the discovery appliance which provided the info for this run.</td>
</tr>
<tr>
<td>Consolidation Data Last Received</td>
<td>_consolidation_last_received date</td>
<td>Date and time last received consolidation data.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>_consolidation list: string</td>
<td>List of the consolidation appliance identifiers this run needs to be sent to.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>_consolidation_done list: string</td>
<td>List of the consolidation appliance identifiers this run has been sent to.</td>
</tr>
<tr>
<td>Multi-tenant company</td>
<td>cdm_company string</td>
<td>Multi-tenant company name.</td>
</tr>
</tbody>
</table>

⚠️ Starting from v10.0 the Discovery Run node has no `run_type` attribute.

### Discovery Run node relationships

The relationships of a Discovery Run node are described in the following table.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Endpoints</td>
<td>DiscoveryRun: List: List: Member: DiscoveryAccess</td>
<td>Discovered Accesses made within this run</td>
</tr>
<tr>
<td>Provider Accesses</td>
<td>DiscoveryRun: List: List: Member: ProviderAccess</td>
<td>Accesses to provider types within this run</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>DiscoveryRun: EndpointRange: EndpointRange: IPRange</td>
<td>IP Range used by this run</td>
</tr>
<tr>
<td>Dropped IPs</td>
<td>DiscoveryRun: EndpointRange: EndpointRange: DroppedEndpoints</td>
<td>Endpoints which were not scanned.</td>
</tr>
</tbody>
</table>

### Discovery Access node

Each time an endpoint is discovered, a Discovery Access node is created. It has relationships to the retrieved discovery information. Discovery Access nodes are created even if discovery fails, so that failed access is clearly identified in the model.

The endpoint attribute of a Discovery Access node describes the target used for discovery. The target for discovery will currently always be an IP v4 or IP v6 address.

### Discovery Access Lifecycle

The following section describes the scenarios in which a Discovery Access is created or destroyed. DDD nodes are never updated.
Creation
A Discovery Access node is created when BMC Atrium Discovery scans an endpoint. By default, a Discovery Access node is always created and stored, whether the discovery succeeded or failed. Where there is no response and there never has been a response from that particular IP address two strategies exist:

- Keep the last one. Each time another no response is detected the new one is created and all previous no response Discovery Access nodes are removed. This is the default behaviour.
- Keep none of them. Each time another no response is detected it is immediately removed. You can choose to use this option in order to improve the quality of the stored data in your system.

Removal
Aging parameters exist for Directly Discovered Data Nodes which have a particular cut-off time. The Aging process will remove any Discovery Access Nodes and all of the other DDD Nodes associated with that node if this default time period is exceeded.

However, if the Discovery Access Node is the only one for an endpoint, then it is not removed. This enables BMC Atrium Discovery to keep a permanent record of every endpoint that has ever been scanned. These Discovery Access Nodes are the only ones which can exceed the default cut-off Aging process. The only time all Discovery Access Nodes are removed for an endpoint is where the endpoint is considered dark space.

The cut-off time is configurable, see DDD removal for details.

Discovery Access node attributes
The attributes and relationships of a Discovery Access node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range Prefix</td>
<td>range_prefix string</td>
<td>Distinguish overlapping address spaces.</td>
</tr>
<tr>
<td>Endpoint</td>
<td>endpoint string</td>
<td>Accessed IP address.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__index int</td>
<td>Index of endpoint for owning range.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>reasoning_uuid string</td>
<td>Reasoning machine that performed discovery.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>engine_id int</td>
<td>ECA engine on Reasoning machine the performed discovery.</td>
</tr>
</tbody>
</table>

State of access — completed, in progress, and so forth.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
<td>state string</td>
<td></td>
</tr>
<tr>
<td>Start Time</td>
<td>starttime date</td>
<td>Date and time access started.</td>
</tr>
<tr>
<td>End Time</td>
<td>endtime date</td>
<td>Date and time access ended.</td>
</tr>
<tr>
<td>Discovery Start Time</td>
<td>discovery_starttime date</td>
<td>Date and time access started on a scanning appliance.</td>
</tr>
<tr>
<td>Discovery End Time</td>
<td>discovery_endtime date</td>
<td>Date and time access ended on a scanning appliance.</td>
</tr>
<tr>
<td>Result</td>
<td>result string</td>
<td>Result of access.</td>
</tr>
<tr>
<td>End State</td>
<td>end_state string</td>
<td>Final state of access.</td>
</tr>
<tr>
<td>Reason</td>
<td>reason string</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td>On Hold Since</td>
<td>is_being_held date</td>
<td>True if access is on hold waiting for a scan window.</td>
</tr>
<tr>
<td>On Hold Duration</td>
<td>on_hold_duration int</td>
<td>How long the access was held waiting for a scan window.</td>
</tr>
<tr>
<td>Not shown in UI</td>
<td>_first_marker boolean</td>
<td>True if this is the first DiscoveryAccess for an endpoint.</td>
</tr>
<tr>
<td>Not shown in UI</td>
<td>_last_marker boolean</td>
<td>True if this is the last DiscoveryAccess for an endpoint.</td>
</tr>
<tr>
<td>Dark Space</td>
<td>dark_space boolean</td>
<td>True if when this was the last DiscoveryAccess for an endpoint it was to be treated as dark space.</td>
</tr>
<tr>
<td>Not shown in UI</td>
<td>_last_interesting boolean</td>
<td>True if this is the last DiscoveryAccess which accessed an endpoint.</td>
</tr>
<tr>
<td>Session Establishment Duration</td>
<td>discovery_duration int</td>
<td>The amount of time taken to establish a session with the endpoint (in seconds).</td>
</tr>
<tr>
<td>Total Discovery Duration</td>
<td>discovery_duration_sum int</td>
<td>Total time in seconds spend in discovery including session establishment time.</td>
</tr>
<tr>
<td>Not shown in UI</td>
<td>access_failure boolean</td>
<td>True if there are session failures linked to this DiscoveryAccess.</td>
</tr>
<tr>
<td>Not shown in UI</td>
<td>access_success string</td>
<td>The session type and credential used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>List of the consolidation appliance identifiers this DA has been be sent to.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>_consolidation_done</td>
<td>list:string</td>
<td>True if present, always have the value True and indicates that the data for this DiscoveryAccess originated from a scanning appliance before being consolidated on this one.</td>
</tr>
<tr>
<td>is_consolidation</td>
<td>boolean</td>
<td>True if present, always have the value True and indicates that the data for this DiscoveryAccess originated from a scanning appliance before being consolidated on this one.</td>
</tr>
<tr>
<td>no_response_count</td>
<td>int</td>
<td>The number of no response DAs suppressed onto a single no response DA.</td>
</tr>
<tr>
<td>last_response</td>
<td>date</td>
<td>The date of the last response DA.</td>
</tr>
<tr>
<td>is_scanner_file</td>
<td>boolean</td>
<td>True if this data came from a scanner file.</td>
</tr>
</tbody>
</table>

**Discovery Atrium Discovery 10.2**

**Node Relationships**

The relationships of a Discovery Access node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>

Network Connections.
<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Connections</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: NetworkConnectionList</td>
<td>HBAInfoList</td>
</tr>
<tr>
<td>HBAs</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: HBAS</td>
<td>HBAs.</td>
</tr>
<tr>
<td>FQDNs</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: FQDNList</td>
<td>FQDNs.</td>
</tr>
<tr>
<td>Files</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: DiscoveredFile</td>
<td>Files.</td>
</tr>
<tr>
<td>Command Results</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: DiscoveredCommandResult</td>
<td>Command results.</td>
</tr>
<tr>
<td>WBEM Instances</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: DiscoveredWBEMEnumInstances</td>
<td>WBEM Instances.</td>
</tr>
<tr>
<td>Registry Queries</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: DiscoveredRegistryValue</td>
<td>Registry Queries.</td>
</tr>
<tr>
<td>SNMP Table Queries</td>
<td>DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: DiscoveredSNMPTable</td>
<td>SNMP Table Query.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Network Device Chassis.</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td></td>
<td>DiscoveredChassisList</td>
<td></td>
</tr>
<tr>
<td>Network Device Cards</td>
<td>DiscoveryAccess: DiscoveryAccess:</td>
<td>Sysplexes.</td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredCardList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredSysplexList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredStorageSubsystemList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredDiskDriveList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredTapeDriveList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredSoftwareList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveredApplicationComponentList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredTransactionList</td>
<td></td>
</tr>
<tr>
<td>Mainframe Programs</td>
<td>DiscoveryAccess: DiscoveryAccess:</td>
<td>Mainframe Programs.</td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredProgramList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredMQDetailList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredDatabaseList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveredDatabaseDetailList</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DiscoveryAccessResult: DiscoveryResult: DiscoveredDependencyList</td>
<td></td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Discovery Run</td>
<td>DiscoveryAccess: Member: List: List: DiscoveryRun</td>
<td>Discovery Run.</td>
</tr>
<tr>
<td>Errors</td>
<td>DiscoveryAccess: ElementWithStatus: Status: Status: ECAError</td>
<td>Errors detected by the ECA engine.</td>
</tr>
<tr>
<td>Optimized Host</td>
<td>DiscoveryAccess: DiscoveryAccess: AccessOptimization: InferredElement: Host</td>
<td>Host corresponding to an access that was optimized away.</td>
</tr>
<tr>
<td>Optimized Network Device</td>
<td>DiscoveryAccess: DiscoveryAccess: AccessOptimization: InferredElement: NetworkDevice</td>
<td>NetworkDevice corresponding to an access that was optimized away.</td>
</tr>
<tr>
<td>Optimized Printer</td>
<td>DiscoveryAccess: DiscoveryAccess: AccessOptimization: InferredElement: Printer</td>
<td>Printer corresponding to an access that was optimized away.</td>
</tr>
<tr>
<td>Optimized SNMP Managed Device</td>
<td>DiscoveryAccess: DiscoveryAccess: AccessOptimization: InferredElement: SNMPManagedDevice</td>
<td>SNMP Managed Device corresponding to an access that was optimized away.</td>
</tr>
</tbody>
</table>
Device Info node

A Device Info node stores information about a scanned IP device on the network. For example, an IP device might be one of the following:

- A host computer (desktop, laptop, server, mainframe, mid-range and so forth). In these cases, there will be a corresponding Host node also.
- A router, switch, load balancer, firewall, IP phone, and so forth.
- A printer.

Device Info node lifecycle

The following section describes the scenarios in which a Discovery Info node is created or destroyed. DDD nodes are never updated.

Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. If there is any response to the scan a Discovery Info node is also created. The level of information in the Device Info node will vary depending on the discovered device.
Removal
A Device Info node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

Device Info node attributes
The attributes of a Device Info node are described in the table below.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Method</td>
<td>Discovery method.</td>
</tr>
<tr>
<td>discovery_method string</td>
<td></td>
</tr>
<tr>
<td>Discovery Duration</td>
<td>Time in seconds spent in discovery.</td>
</tr>
<tr>
<td>discovery_duration int</td>
<td></td>
</tr>
<tr>
<td>Request Time</td>
<td>When this request was made.</td>
</tr>
<tr>
<td>request_time date</td>
<td></td>
</tr>
<tr>
<td>Failure Reason</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td>failure_reason string</td>
<td></td>
</tr>
<tr>
<td>Hostname</td>
<td>Name of host.</td>
</tr>
<tr>
<td>hostname string</td>
<td></td>
</tr>
<tr>
<td>Fully Qualified Domain Names</td>
<td>Fully qualified domain name of the host.</td>
</tr>
<tr>
<td>fqdn string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS</td>
<td>Operating system.</td>
</tr>
<tr>
<td>os string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Class</td>
<td>Operating system class.</td>
</tr>
<tr>
<td>os_class string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Type</td>
<td>Operating system type.</td>
</tr>
<tr>
<td>os_type string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Version</td>
<td>Operating system version.</td>
</tr>
<tr>
<td>os_version string</td>
<td></td>
</tr>
<tr>
<td>Service Pack</td>
<td>Service pack details.</td>
</tr>
<tr>
<td>service_pack int</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Architecture</td>
<td>Operating system architecture.</td>
</tr>
<tr>
<td>os_arch string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS System Directory</td>
<td>Operating system directory.</td>
</tr>
<tr>
<td>os_directory string</td>
<td></td>
</tr>
<tr>
<td>os_edition string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Update Level</td>
<td>Operating system update level.</td>
</tr>
<tr>
<td>os_level string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Build</td>
<td>Operating system build information.</td>
</tr>
<tr>
<td>os_build string</td>
<td></td>
</tr>
<tr>
<td>Discovered OS Vendor</td>
<td>Operating system vendor.</td>
</tr>
<tr>
<td>os_vendor string</td>
<td></td>
</tr>
<tr>
<td>Vendor of device</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>Vendor</td>
<td>Vendor string</td>
</tr>
<tr>
<td>Platform</td>
<td>Platform scripts used in discovery.</td>
</tr>
<tr>
<td>Model</td>
<td>Model of device.</td>
</tr>
<tr>
<td>DNS Domain</td>
<td>DNS domain.</td>
</tr>
<tr>
<td>NIS/Windows Domain</td>
<td>Windows domain.</td>
</tr>
<tr>
<td>Device Type</td>
<td>Device type.</td>
</tr>
<tr>
<td>Last Access Method</td>
<td>The last method used to access this device.</td>
</tr>
<tr>
<td>Login Authentication Method</td>
<td>How the login session was authenticated - may be 'key' or 'password'.</td>
</tr>
<tr>
<td>Last Credential ID</td>
<td>The ID of the last credential used to access this device.</td>
</tr>
<tr>
<td>Last Windows Proxy Pool</td>
<td>The name of the last Windows Proxy Pool used to access this device.</td>
</tr>
<tr>
<td>Last Windows Proxy</td>
<td>The name of the last Windows Proxy used to access this device.</td>
</tr>
<tr>
<td>Other Credentials</td>
<td>List of other credential IDs used to access this device.</td>
</tr>
<tr>
<td>SNMP sysname</td>
<td>SNMP sysname.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of device.</td>
</tr>
<tr>
<td>Capability Ids</td>
<td>Capability Ids.</td>
</tr>
<tr>
<td>Capability Types</td>
<td>Capability Types.</td>
</tr>
<tr>
<td>SNMP sysObjectld</td>
<td>SNMP sysObjectld.</td>
</tr>
<tr>
<td>SNMP sysDescr</td>
<td>SNMP sysDescr.</td>
</tr>
<tr>
<td>Serial number</td>
<td>Serial number.</td>
</tr>
<tr>
<td>SNMP syscontact</td>
<td>SNMP syscontact.</td>
</tr>
<tr>
<td>SNMP syslocation</td>
<td>SNMP syslocation.</td>
</tr>
</tbody>
</table>
### SNMP cdpGlobalDeviceId

| cdpdeviceid string | SNMP cdpGlobalDeviceId. |

### Nexus VDC ID

| nexus_vdc_id string | Cisco Nexus VDC ID. |

### SNMP v3 Engine Identifier

| snmpv3_engine_id string | SNMP v3 Engine Identifier. |

### Operating system derived from discovery heuristics

| probed_os string | Operating system derived from discovery heuristics. |

### Operating system type derived from discovery heuristics

| probed_os_type string | Operating system type derived from discovery heuristics. |

### Operating system version derived from discovery heuristics

| probed_os_version string | Operating system version derived from discovery heuristics. |

### CVE-2011-1785 Vulnerability

| _cve_2011_1785 _boolean | CVE-2011-1785 vulnerability flag (vSphere based ESX/ESXi hosts only). |

### Firmware Version

| firmware_version string | Firmware version. |

### Testing Status

| testing_status string | Testing status for non Host devices. |

### Processing Messages

| processing_messages list:string | Optional list of processing messages. |

### Not displayed in UI

| command_status_failure boolean | Flag that this node has command failures linked to it. |

### Not displayed in UI

| method_failure boolean | Flag that this node has script failures linked to it. |

### Not displayed in UI

| method_success string | The name of the script that succeeded in getting this data. |

### Access Method

| access_method string | The access method used by the script that succeeded in getting this data. |

## Device Info node relationships

The relationships of a Device Info node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requesting Pattern</td>
<td>DeviceInfo: DiscoveryResult: Request: Pattern: Pattern</td>
<td>Pattern from which the request to retrieve this information originate.</td>
</tr>
<tr>
<td>Host</td>
<td>DeviceInfo: Primary: Inference: InferredElement: Host</td>
<td>Inferred Host.</td>
</tr>
<tr>
<td>Mainframe</td>
<td></td>
<td>Inferred Mainframe.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>Storage Device</td>
<td>DeviceInfo: Primary: Inference: InferredElement: StorageDevice</td>
<td>Inferred Storage Device.</td>
</tr>
<tr>
<td>Printer</td>
<td>DeviceInfo: Primary: Inference: InferredElement: Printer</td>
<td>Inferred Printer.</td>
</tr>
<tr>
<td>SNMP Managed Device</td>
<td>DeviceInfo: Primary: Inference: InferredElement: SNMPManagedDevice</td>
<td>Inferred SNMP Managed Device.</td>
</tr>
</tbody>
</table>

**Host Info node**

A Host Info node contains information about the kind of host and some of its physical characteristics. If BMC Atrium Discovery decides that there is not sufficient information available to create a Host node, the Host Info node captures and stores the available information.

**Host Info Lifecycle**

The following section describes the scenarios in which a Host Info node is created or destroyed. DDD nodes are never updated.

**Creation**

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. If there is any response to the scan a Host Info node is also created. The level of information in the Host Info node varies depending on the discovered device.

**Removal**

A Host Info node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

**Host Info node attributes**

The attributes and relationships of a Host Info node are described in the table below.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type or Relationship Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Method</td>
<td>discovery_method string</td>
<td>Discovery method.</td>
</tr>
<tr>
<td>Discovery Duration</td>
<td>discovery_duration int</td>
<td>Time in seconds spent in discovery.</td>
</tr>
<tr>
<td>Request Time</td>
<td>request_time date</td>
<td>When this request was made.</td>
</tr>
<tr>
<td>Failure Reason</td>
<td>failure_reason string</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td>Hostid</td>
<td>hostid string</td>
<td>Unique Host identification string.</td>
</tr>
<tr>
<td>Uptime Days</td>
<td>uptime int</td>
<td>The time in days since the host was booted.</td>
</tr>
<tr>
<td>Uptime Seconds</td>
<td>uptimeSeconds int</td>
<td>The time in seconds since the host was booted.</td>
</tr>
<tr>
<td>DF Output</td>
<td>df string</td>
<td>Output from UNIX df command.</td>
</tr>
<tr>
<td>Discovered Kernel</td>
<td>kernel string</td>
<td>Kernel details.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>probed_os string</td>
<td>Operating system derived from discovery heuristics.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>probed_os_type string</td>
<td>Operating system type derived from discovery heuristics.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>probed_os_version string</td>
<td>Operating system version derived from discovery heuristics.</td>
</tr>
<tr>
<td>Number of Processors</td>
<td>num_processors int</td>
<td>The number of physical processors.</td>
</tr>
<tr>
<td>Processor Type</td>
<td>processor_type string</td>
<td>The type of each processor.</td>
</tr>
<tr>
<td>Processor Speed</td>
<td>processor_speed int</td>
<td>The speed of each processor in MHz.</td>
</tr>
<tr>
<td>Number of Logical Processors</td>
<td>num_logical_processors int</td>
<td>The number of logical processors available to the OS.</td>
</tr>
<tr>
<td>Cores per Processor</td>
<td>cores_per_processor int</td>
<td>The number of cores per physical processor available to the OS.</td>
</tr>
<tr>
<td>Threads per Processor Core</td>
<td>threads_per_core int</td>
<td>The number of threads per core in multi/hyper threaded processors available to the OS.</td>
</tr>
<tr>
<td>CPU Threading Enabled</td>
<td>cpu_threading_enabled boolean</td>
<td>Whether CPU hardware threading is enabled.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The number of physical processor types.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type or Relationship Target</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Number of Processor Types</td>
<td>num_processor_types int</td>
<td></td>
</tr>
<tr>
<td>All Processor Types</td>
<td>all_processor_types _list:string</td>
<td>List of all processor types.</td>
</tr>
<tr>
<td>All Processor Speeds</td>
<td>all_processor_speeds _list:int</td>
<td>List of all processor speeds.</td>
</tr>
<tr>
<td>Logical Processor Counts</td>
<td>all_processor_logical_counts _list:int</td>
<td>List of logical processor counts.</td>
</tr>
<tr>
<td>Physical Processor Counts</td>
<td>all_processor_physical_counts _list:int</td>
<td>List of physical processor counts.</td>
</tr>
<tr>
<td>Logical RAM</td>
<td>logical_ram int</td>
<td>Amount of RAM (in MB) available on the host, as reported by OS.</td>
</tr>
<tr>
<td>Physical RAM</td>
<td>ram int</td>
<td>The amount of RAM (in MB) installed on the host.</td>
</tr>
<tr>
<td>E10K SSP Hostname</td>
<td>ssphostname string</td>
<td>E10K SSP Hostname.</td>
</tr>
<tr>
<td>SunFire Domain</td>
<td>sunfire_domain string</td>
<td>F15K SunFire domain.</td>
</tr>
<tr>
<td>Zonename</td>
<td>zonename string</td>
<td>Solaris 10 Zonename.</td>
</tr>
<tr>
<td>Windows Workgroup</td>
<td>workgroup string</td>
<td>Windows workgroup.</td>
</tr>
<tr>
<td>Hardware Vendor</td>
<td>vendor string</td>
<td>Hardware vendor.</td>
</tr>
<tr>
<td>Model</td>
<td>model string</td>
<td>Hardware model.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>serial string</td>
<td>Serial number.</td>
</tr>
<tr>
<td>Windows UUID</td>
<td>windows_uuid string</td>
<td>Windows UUID.</td>
</tr>
<tr>
<td>Power Supply Status</td>
<td>psu_status list:string</td>
<td>Power Supply Unit status.</td>
</tr>
<tr>
<td>Cluster Instance ID</td>
<td>cluster_instance_id string</td>
<td>Cluster instance identifier.</td>
</tr>
<tr>
<td>Cluster Name</td>
<td>cluster_name string</td>
<td>Cluster name.</td>
</tr>
<tr>
<td>Cluster Name Resource</td>
<td>cluster_name_resource string</td>
<td>Cluster name resource.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LPAR Active CPUs In Pool</td>
<td>lpar_active_cpus_in_pool int</td>
<td>The maximum number of CPUs available to this LPAR's shared processor pool.</td>
</tr>
<tr>
<td>LPAR Active Physical CPUs In System</td>
<td>lpar_active_physical_cpus_in_system int</td>
<td>The current number of active physical CPUs in the system containing this LPAR.</td>
</tr>
<tr>
<td>LPAR Capacity Increment</td>
<td>lpar_capacity_increment float</td>
<td>The granule at which changes to Entitled Capacity can be made. A value in whole multiples indicates a Dedicated LPAR.</td>
</tr>
<tr>
<td>LPAR Entitled Capacity</td>
<td>lpar_entitled_capacity float</td>
<td>The number of processing units this LPAR is entitled to receive.</td>
</tr>
<tr>
<td>LPAR Entitled Capacity Of Pool</td>
<td>lpar_entitled_capacity_of_pool float</td>
<td>The number of processing units that this LPAR's shared processor pool is entitled to receive.</td>
</tr>
<tr>
<td>LPAR Maximum Capacity</td>
<td>lpar_maximum_capacity float</td>
<td>The maximum number of processing units this LPAR was defined to ever have. Entitled capacity can be increased up to this value.</td>
</tr>
<tr>
<td>LPAR Maximum Capacity Of Pool</td>
<td>lpar_maximum_capacity_of_pool int</td>
<td>The maximum number of processing units available to this LPAR's shared processor pool.</td>
</tr>
<tr>
<td>LPAR Maximum Memory</td>
<td>lpar_maximum_memory int</td>
<td>Maximum possible amount of memory.</td>
</tr>
<tr>
<td>LPAR Maximum Physical CPUs In System</td>
<td>lpar_maximum_physical_cpus_in_system int</td>
<td>The maximum possible number of physical CPUs in the system containing this LPAR.</td>
</tr>
<tr>
<td>LPAR Maximum Virtual CPUs</td>
<td>lpar_maximum_virtual_cpus int</td>
<td>Maximum possible number of CPUs (virtual engines).</td>
</tr>
<tr>
<td>LPAR Minimum Capacity</td>
<td>lpar_minimum_capacity float</td>
<td>The minimum number of processing units this LPAR was defined to ever have. Entitled capacity can be reduced down to this value.</td>
</tr>
<tr>
<td>LPAR Minimum Memory</td>
<td>lpar_minimum_memory int</td>
<td>Minimum memory this LPAR was defined to ever have.</td>
</tr>
<tr>
<td>LPAR Minimum Virtual CPUs</td>
<td>lpar_minimum_virtual_cpus int</td>
<td>Minimum number of virtual CPUs this LPAR was defined to ever have.</td>
</tr>
<tr>
<td>LPAR Mode</td>
<td>lpar_mode string</td>
<td>Indicates whether the LPAR processor capacity is capped, or if it is uncapped and allowed to consume idle cycles from the shared pool. Dedicated LPAR is capped or donating.</td>
</tr>
<tr>
<td>LPAR Node Name</td>
<td>lpar_node_name string</td>
<td>Hostname of the LPAR.</td>
</tr>
<tr>
<td>LPAR Online Memory</td>
<td>lpar_online_memory int</td>
<td>The amount of memory currently online (in MB).</td>
</tr>
<tr>
<td>LPAR Online Virtual CPUs</td>
<td>lpar_online_virtual_cpus int</td>
<td>Number of CPUs (virtual engines) currently online.</td>
</tr>
<tr>
<td>LPAR Group Identifier</td>
<td>lpar_partition_group_id int</td>
<td>LPAR group that this LPAR is a member of.</td>
</tr>
<tr>
<td></td>
<td>lpar_partition_group_id int</td>
<td>Logical partition name as assigned by the HMC.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type or Relationship Target</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LPAR Name</td>
<td>lpar_partition_name string</td>
<td></td>
</tr>
<tr>
<td>LPAR Identifier</td>
<td>lpar_partition_number int</td>
<td>Number of the logical partition.</td>
</tr>
<tr>
<td>LPAR Physical CPU Percentage</td>
<td>lpar_physical_cpu_percentage float</td>
<td>Fractional representation relative to whole physical CPUs that these LPARs virtual CPUs equate to. This is a function of Entitled Capacity / Online CPUs. Dedicated LPARs would have 100% Physical CPU Percentage. A 4-way virtual with Entitled Capacity of 2 processor units would have a 50% physical CPU Percentage.</td>
</tr>
<tr>
<td>LPAR Shared Physical CPUs In System</td>
<td>lpar_shared_physical_cpus_in_system int</td>
<td>The number of physical CPUs available for use by shared processor LPARs.</td>
</tr>
<tr>
<td>LPAR Shared Pool Identifier</td>
<td>lpar_shared_pool_id int</td>
<td>Identifier of Shared Pool of Physical processors that this LPAR is a member.</td>
</tr>
<tr>
<td>LPAR Type</td>
<td>lpar_type string</td>
<td>LPAR type: dedicated or shared, SMT enabled or not.</td>
</tr>
<tr>
<td>LPAR Unallocated Capacity</td>
<td>lpar_unallocated_capacity float</td>
<td>The sum of the number of processor units unallocated from shared LPARs in an LPAR group. This sum does not include the processor units unallocated from a dedicated LPAR, which can also belong to the group. The unallocated processor units can be allocated to any dedicated LPAR (if it is greater than or equal to 1.0) or shared LPAR of the group.</td>
</tr>
<tr>
<td>LPAR Unallocated Weight</td>
<td>lpar_unallocated_weight int</td>
<td>Number of variable processor capacity weight units currently unallocated within the LPAR group.</td>
</tr>
<tr>
<td>LPAR Variable Capacity Weight</td>
<td>lpar_variable_capacity_weight int</td>
<td>The priority weight assigned to this LPAR which controls how extra (idle) capacity is allocated to it. A weight of -1 indicates a soft cap is in place.</td>
</tr>
<tr>
<td>WPAR Identifier</td>
<td>wparid int</td>
<td>The identifier (key) of the AIX WPAR. An identifier of 0 indicates that the system is the WPAR container.</td>
</tr>
<tr>
<td>System Identifier</td>
<td>systemid string</td>
<td>An identifier that links all the AIX/VIO LPARs in a physical system (including WPAs).</td>
</tr>
<tr>
<td>LPAR Desired Entitled Capacity</td>
<td>lpar_desired_capacity float</td>
<td>Desired entitled CPU capacity for LPAR.</td>
</tr>
<tr>
<td>LPAR Desired Memory</td>
<td>lpar_desired_memory int</td>
<td>Desired amount of main memory for LPAR in MB.</td>
</tr>
<tr>
<td>LPAR Desired CPUs</td>
<td>lpar_desired_cpus int</td>
<td>Desired number of virtual processors for LPAR.</td>
</tr>
<tr>
<td>LPAR Minimum Capacity per Virtual Processor</td>
<td>lpar_minimum_capacity_per_vp float</td>
<td>Minimum entitled capacity required per virtual processor for LPAR.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>command_status_failure boolean</td>
<td>Flag that this node has command failures linked to it.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>method_failure boolean</td>
<td>Flag that this node has script failures linked to it.</td>
</tr>
</tbody>
</table>
UI Name | Attribute Name and Type or Relationship Target | Description
--- | --- | ---
 | method_success string | The name of the script that succeeded in getting this data.
 | access_method string | The access method used by the script that succeeded in getting this data.

### Host Info node relationships

The relationships of a Host Info node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requesting Pattern</td>
<td>DiscoveryResult:Request:Pattern:Pattern</td>
<td>Pattern from which the request to retrieve this information originate.</td>
</tr>
<tr>
<td>Host</td>
<td>Primary:Inference:InferredElement:Host</td>
<td>Inferred Host.</td>
</tr>
<tr>
<td>Script Command Errors</td>
<td>DiscoveryResult:Metadata:Detail:CommandFailure</td>
<td>Script command errors.</td>
</tr>
<tr>
<td>Script Execution Errors</td>
<td>DiscoveryResult:Metadata:Detail:ScriptFailure</td>
<td>Script execution errors.</td>
</tr>
</tbody>
</table>

### Directory Listing node

A Directory Listing node stores information about a list of directory entries.

### Directory Listing Lifecycle

The following section describes the scenarios in which a Directory Listing node is created or destroyed. DDD nodes are never updated.

#### Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A directory listing can be obtained using the `discovery.listDirectory` function. A Directory Listing node is created for the listing.

#### Removal

A Directory Listing node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

### Directory Listing node attributes

The attributes and relationships of a Directory Listing node are described in the table below.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type or Relationship Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directory Name</td>
<td>path string</td>
<td>Path of the directory.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>system _boolean</td>
<td>Flag if request was made by system pattern.</td>
</tr>
<tr>
<td>Check Windows Redirect Location</td>
<td>win64_redirect boolean</td>
<td>Look in both redirected and original file location on 64-bit Windows.</td>
</tr>
<tr>
<td>Retrieved Path</td>
<td>actual_path string</td>
<td>Actual file location (if different to path).</td>
</tr>
<tr>
<td>Discovery Method</td>
<td>discovery_method string</td>
<td>Discovery method.</td>
</tr>
<tr>
<td>Discovery Duration</td>
<td>discovery_duration int</td>
<td>Time in seconds spent in discovery.</td>
</tr>
<tr>
<td>Request Time</td>
<td>request_time date</td>
<td>When this request was made.</td>
</tr>
<tr>
<td>Failure Reason</td>
<td>failure_reason string</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>command_status_failure boolean</td>
<td>Flag that this node has command failures linked to it.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>method_failure boolean</td>
<td>Flag that this node has script failures linked to it.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>method_success string</td>
<td>The name of the script that succeeded in getting this data.</td>
</tr>
<tr>
<td>Access Method</td>
<td>access_method string</td>
<td>The access method used by the script that succeeded in getting this data.</td>
</tr>
</tbody>
</table>

**Directory Listing node relationships**

The relationships of a Directory List node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requesting Pattern</td>
<td>DirectoryListing: DiscoveryResult: Request: Pattern: Pattern</td>
<td>Pattern from which the request to retrieve this information originate.</td>
</tr>
</tbody>
</table>
Registry Listing node

A Registry Listing node stores information about a list of directory entries.

Registry Listing Lifecycle

The following section describes the scenarios in which a Registry Listing node is created or destroyed. DDD nodes are never updated.

Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A registry listing can be obtained using the `discovery.listRegistry` function. A Registry Listing node is created for the listing.

Removal

A Registry Listing node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

Registry Listing node attributes

The attributes and relationships of a Registry Listing node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query</td>
<td>query string</td>
<td>Registry query.</td>
</tr>
<tr>
<td></td>
<td>system_boolean</td>
<td>Flag if request was made by system pattern.</td>
</tr>
<tr>
<td>Check Windows Redirect Location</td>
<td>win64_redirect boolean</td>
<td>Look in both redirected and original registry location on 64-bit Windows.</td>
</tr>
<tr>
<td>Retrieved Query</td>
<td>actual_query string</td>
<td>Actual registry location (if different to query).</td>
</tr>
<tr>
<td>Discovery Method</td>
<td>discovery_method string</td>
<td>Discovery method.</td>
</tr>
<tr>
<td>Discovery Duration</td>
<td>discovery_duration int</td>
<td>Time in seconds spent in discovery.</td>
</tr>
<tr>
<td>Request Time</td>
<td>request_time date</td>
<td>When this request was made.</td>
</tr>
<tr>
<td>Failure Reason</td>
<td>failure_reason string</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Flag that this node has command failures linked to it.</td>
</tr>
</tbody>
</table>
### UI Name
**Attribute Name and Type**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not displayed in UI</td>
</tr>
<tr>
<td>command_status_failure boolean</td>
</tr>
<tr>
<td>Not displayed in UI</td>
</tr>
<tr>
<td>method_failure boolean</td>
</tr>
<tr>
<td>Not displayed in UI</td>
</tr>
<tr>
<td>method_success string</td>
</tr>
<tr>
<td>Access Method</td>
</tr>
<tr>
<td>access_method string</td>
</tr>
</tbody>
</table>

### Registry Listing node relationships

The relationships of a Registry List node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entries</td>
<td>RegistryListing: List: List: Member: DiscoveredRegistryEntry</td>
<td>Entries in the registry.</td>
</tr>
<tr>
<td>Requesting Pattern</td>
<td>RegistryListing: DiscoveryResult: Request: Pattern: Pattern</td>
<td>Pattern from which the request to retrieve this information originated.</td>
</tr>
</tbody>
</table>

### Service List node

A Service List node stores information about a list of Windows services running on a host.

### Service List Lifecycle

The following section describes the scenarios in which a Service List node is created or destroyed. DDD nodes are never updated.

#### Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A Service List node is created for the listing.

#### Removal

A Service List node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.
Service List node attributes

The attributes and relationships of a Service List node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery Method</td>
<td>discovery_method string</td>
<td>The name of the discovery method used to connect to this host.</td>
</tr>
<tr>
<td>_system</td>
<td></td>
<td>Flag if request was made by system pattern.</td>
</tr>
<tr>
<td>Discovery duration</td>
<td>discovery_duration int</td>
<td>Time spent in discovery (seconds).</td>
</tr>
<tr>
<td>Request time</td>
<td>request_time date</td>
<td>The time that this request was made.</td>
</tr>
<tr>
<td>Failure reason</td>
<td>failure_reason string</td>
<td>Reason for failure, if any.</td>
</tr>
<tr>
<td>command_status_failure</td>
<td>boolean</td>
<td>Flag showing whether this node has command failures linked to it.</td>
</tr>
<tr>
<td>method_failure</td>
<td>boolean</td>
<td>Flag showing whether this node has script failures linked to it.</td>
</tr>
<tr>
<td>method_success</td>
<td>string</td>
<td>The name of the script that succeeded in getting this data.</td>
</tr>
<tr>
<td>access_method</td>
<td>string</td>
<td>The access method used by the script that succeeded in getting this data.</td>
</tr>
</tbody>
</table>

Service List node relationships

The relationships of a Service List node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Services</td>
<td>ServiceList: List: List: Member: DiscoveredService</td>
<td>Services in the list.</td>
</tr>
<tr>
<td>Requesting Pattern</td>
<td>ServiceList: DiscoveryResult: Request: Pattern: Pattern</td>
<td>Pattern from which the request to retrieve this information originated.</td>
</tr>
</tbody>
</table>

Discovered Directory Entry node

A Discovered Directory Entry node stores information about an entry in a directory listing.
Discovered Directory Entry Lifecycle

The following section describes the scenarios in which a Discovered Directory Entry node is created or destroyed. DDD nodes are never updated.

Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A directory listing can be obtained using the `discovery.listDirectory` function. A Discovered Directory Entry node is created for each file in the listing.

Removal

A Discovered Directory Entry node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

Discovered Directory Entry node attributes

The attributes and relationships of a Discovered Directory Entry node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type or Relationship Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>File Name.</td>
</tr>
<tr>
<td>Full Path</td>
<td>path string</td>
<td>Full path to the file including drive letter (Windows).</td>
</tr>
<tr>
<td>Drive</td>
<td>drive string</td>
<td>Drive letter (Windows).</td>
</tr>
<tr>
<td>DOS 8.3 File Name</td>
<td>dos_name string</td>
<td>DOS 8.3 format file name (Windows).</td>
</tr>
<tr>
<td>File Type</td>
<td>file_type string</td>
<td>File Type. Possible file types are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Directory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• File</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Symbolic Link</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Character Device</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Block Device</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Socket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FIFO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Solaris Door</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• HP-UX Network Special</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BSD UnionFS White Out</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• XENIX Semaphore</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• XENIX Shared Memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unknown</td>
</tr>
<tr>
<td>Links To</td>
<td>link string</td>
<td>Path to symbolically linked file (UNIX Symbolic Links only).</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type or Relationship Target</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>Size</td>
<td>size int</td>
<td>Size of the file.</td>
</tr>
<tr>
<td>Major Device Number</td>
<td>major int</td>
<td>Major device number (UNIX Devices only).</td>
</tr>
<tr>
<td>Minor Device Number</td>
<td>minor int</td>
<td>Minor device number (UNIX Devices only).</td>
</tr>
<tr>
<td>Owner</td>
<td>owner string</td>
<td>Name of the owner of the file.</td>
</tr>
<tr>
<td>Group</td>
<td>group string</td>
<td>Name of the group of the file (UNIX).</td>
</tr>
<tr>
<td>Permissions</td>
<td>permissions list:string</td>
<td>List of file permissions. Possible permissions are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OwnerRead</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OwnerWrite</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OwnerExecute (OwnerSearch for Directories)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupRead</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Group Write</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupExecute (GroupSearch for Directories)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WorldRead</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WorldWrite</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WorldExecute (WorldSearch for Directories)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MandatoryLocking (Solaris only)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SetUID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SetGID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Sticky</td>
</tr>
<tr>
<td>UNIX Permissions String</td>
<td>permissions_string string</td>
<td>UNIX Permissions string (UNIX).</td>
</tr>
<tr>
<td>Mode</td>
<td>mode string</td>
<td>UNIX File Mode (UNIX, Octal format).</td>
</tr>
<tr>
<td>Last Modified</td>
<td>last_modified string</td>
<td>Last modified date/time - platform specific format.</td>
</tr>
<tr>
<td>Hidden File</td>
<td>mode boolean</td>
<td>Indicates a Hidden file (Windows).</td>
</tr>
<tr>
<td>System File</td>
<td>system boolean</td>
<td>Indicates a System file (Windows).</td>
</tr>
</tbody>
</table>

**Discovered Directory Entry node relationships**

The relationships of a Discovered Directory Entry node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directory Listing</td>
<td>DiscoveredDirectoryEntry: Member: List: List: DirectoryListing</td>
<td>Directory listing that this entry is part of.</td>
</tr>
</tbody>
</table>
Discovered Registry Entry node

A Discovered Registry Entry node stores information about an entry in a registry listing.

Discovered Registry Entry Lifecycle

The following section describes the scenarios in which a Discovered Registry Entry node is created or destroyed. DDD nodes are never updated.

Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A registry listing can be obtained using the `discovery.listRegistry` function. A Discovered Registry Entry node is created for each discovered registry key.

Removal

A Discovered Registry Entry node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

Discovered Registry Entry node attributes

The attributes and relationships of a Discovered Registry Entry node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name name string</td>
<td>The name of the registry key.</td>
</tr>
<tr>
<td>Type key_type string</td>
<td>The registry key type. This can be <code>VALUE</code> or <code>SUBKEY</code>.</td>
</tr>
<tr>
<td>Data type data_type string</td>
<td>The registry key data type. For example, <code>REG_SZ</code>, <code>REG_DWORD</code>, and so on.</td>
</tr>
</tbody>
</table>

Discovered Registry Entry node relationships

The relationships of a Discovered Registry Entry node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created Software Instances</td>
<td>DiscoveredDirectoryEntry: Primary:</td>
<td>Software instance whose existence was inferred from this directory entry.</td>
</tr>
<tr>
<td></td>
<td>Inference: InferredElement: SoftwareInstance</td>
<td></td>
</tr>
<tr>
<td>Contributing to Software Instances</td>
<td>DiscoveredDirectoryEntry: Contributor:</td>
<td>Software instance whose attributes have been partly or wholly determined from this directory entry.</td>
</tr>
<tr>
<td></td>
<td>Inference: InferredElement: SoftwareInstance</td>
<td></td>
</tr>
<tr>
<td>Associated to Software Instances</td>
<td>DiscoveredDirectoryEntry: Associate:</td>
<td>Software instance related in some way to this directory entry.</td>
</tr>
<tr>
<td></td>
<td>Inference: InferredElement: SoftwareInstance</td>
<td></td>
</tr>
</tbody>
</table>
## Discovered Service node

A Discovered Service node stores information about an entry in a registry listing.

### Discovered Service Lifecycle

The following section describes the scenarios in which a Discovered Service node is created or destroyed. DDD nodes are never updated.

#### Creation

Whenever BMC Atrium Discovery scans an IP address, a Discovery Access node is created. A Discovered Service node is created for each discovered service.

#### Removal

A Discovered Service node is removed when the Discovery Access node which it is associated with has been destroyed through the Aging process.

### Discovered Service node attributes

The attributes and relationships of a Discovered Service node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name of the service.</td>
</tr>
<tr>
<td>Display Name</td>
<td>display_name int</td>
<td>The user friendly name of the service.</td>
</tr>
<tr>
<td>State</td>
<td>state string</td>
<td>The service state. This can be Started, Stopped, or Paused.</td>
</tr>
<tr>
<td>Process Identifier</td>
<td>pid int</td>
<td>The process identifier.</td>
</tr>
<tr>
<td>Start Mode</td>
<td>start_mode int</td>
<td>The service start mode. This can be Automatic, Manual, or Disabled.</td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Service Runs As</td>
<td>username int</td>
<td>The user name that the service is running as.</td>
</tr>
</tbody>
</table>

### Discovered Service node relationships

The relationships of a Discovered Service node are described in the table below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceList</td>
<td>ServiceList: Primary: Inference: InferredElement: SoftwareInstance</td>
<td>Service listing this service is part of.</td>
</tr>
<tr>
<td>Created Software Instances</td>
<td>ServiceList: Member: List: List: ServiceList</td>
<td>Software instance whose existence was inferred from this service.</td>
</tr>
<tr>
<td>Contributing To Software Instances</td>
<td>ServiceList: Contributor: Inference: InferredElement: SoftwareInstance</td>
<td>Software instance whose attributes have been partly or wholly determined from this service.</td>
</tr>
<tr>
<td>Associated To Software Instances</td>
<td>ServiceList: Associate: Inference: InferredElement: SoftwareInstance</td>
<td>Software instance related in some way to this service.</td>
</tr>
<tr>
<td>Created Software Components</td>
<td>DiscoveredService: Primary: Inference: InferredElement: SoftwareComponent</td>
<td>Software component whose existence was inferred from this service.</td>
</tr>
<tr>
<td>Contributing To Software Components</td>
<td>DiscoveredService: Contributor: Inference: InferredElement: SoftwareComponent</td>
<td>Software component whose attributes have been partly or wholly determined from this service.</td>
</tr>
<tr>
<td>Associated To Software Components</td>
<td>DiscoveredService: Associate: Inference: InferredElement: SoftwareComponent</td>
<td>Software component related in some way to this service.</td>
</tr>
<tr>
<td>Created Runtime Environments</td>
<td>DiscoveredService: Primary: Inference: InferredElement: RuntimeEnvironment</td>
<td>Runtime environment whose existence was inferred from this service.</td>
</tr>
<tr>
<td>Contributing To Runtime Environments</td>
<td>DiscoveredService: Contributor: Inference: InferredElement: RuntimeEnvironment</td>
<td>Runtime environment whose attributes have been partly or wholly determined from this service.</td>
</tr>
<tr>
<td>Associated To Runtime Environments</td>
<td>DiscoveredService: Associate: Inference: InferredElement: RuntimeEnvironment</td>
<td>Runtime environment related in some way to this service.</td>
</tr>
</tbody>
</table>
Other DDD nodes

The remaining DDD nodes are all dependent on the Discovery Access node. These nodes are all created and removed in the same way as previously described:

<table>
<thead>
<tr>
<th>DirectlyDiscoveredData</th>
<th>CommandFailure</th>
<th>DeviceInfo</th>
</tr>
</thead>
<tbody>
<tr>
<td>DirectlyDiscoveredList</td>
<td>DirectoryListing</td>
<td>DiscoveredATMVirtualCircuitList</td>
</tr>
<tr>
<td>DiscoveredATMVirtualPathList</td>
<td>DiscoveredAggregatedPortsList</td>
<td>DiscoveredApplicationComponentList</td>
</tr>
<tr>
<td>DiscoveredBridgeList</td>
<td>DiscoveredCardList</td>
<td>DiscoveredChassisList</td>
</tr>
<tr>
<td>DiscoveredCouplingFacilityList</td>
<td>DiscoveredDatabaseDetailList</td>
<td>DiscoveredDatabaseList</td>
</tr>
<tr>
<td>DiscoveredDependencyList</td>
<td>DiscoveredDevicePortList</td>
<td>DiscoveredDiskDriveList</td>
</tr>
<tr>
<td>DiscoveredFrameRelayDLCILList</td>
<td>DiscoveredFrameRelayLMIList</td>
<td>DiscoveredMFPartList</td>
</tr>
<tr>
<td>DiscoveredMQDetailList</td>
<td>DiscoveredMainframeList</td>
<td>DiscoveredNeighbours</td>
</tr>
<tr>
<td>DiscoveredPackages</td>
<td>DiscoveredPatches</td>
<td>DiscoveredSNMPTable</td>
</tr>
<tr>
<td>DiscoveredSoftwareList</td>
<td>DiscoveredStorageSubsystemList</td>
<td>DiscoveredSysplexList</td>
</tr>
<tr>
<td>DiscoveredTapeDriveList</td>
<td>DiscoveredTransactionList</td>
<td>DiscoveredWBEMAssociators</td>
</tr>
<tr>
<td>DiscoveredWBEMEnumInstances</td>
<td>DiscoveredWMIQuery</td>
<td>FQDNList</td>
</tr>
<tr>
<td>FileSystemList</td>
<td>HBAInfoList</td>
<td>IntegrationResult</td>
</tr>
<tr>
<td>InterfaceList</td>
<td>NetworkConnectionList</td>
<td>ProcessList</td>
</tr>
<tr>
<td>RegistryListing</td>
<td>ServiceList</td>
<td>VirtualMachineList</td>
</tr>
<tr>
<td>DirectlyDiscoveredListMember</td>
<td>DiscoveredATMVirtualCircuit</td>
<td>DiscoveredATMVirtualPath</td>
</tr>
<tr>
<td>DiscoveredAggregatedPorts</td>
<td>DiscoveredApplicationComponent</td>
<td>DiscoveredBridge</td>
</tr>
<tr>
<td>DiscoveredCard</td>
<td>DiscoveredChassis</td>
<td>DiscoveredCouplingFacility</td>
</tr>
<tr>
<td>DiscoveredDatabase</td>
<td>DiscoveredDatabaseDetail</td>
<td>DiscoveredDependency</td>
</tr>
<tr>
<td>DiscoveredDevicePort</td>
<td>DiscoveredDirectoryEntry</td>
<td>DiscoveredDiskDrive</td>
</tr>
<tr>
<td>DiscoveredFQDN</td>
<td>DiscoveredFileSystem</td>
<td>DiscoveredFrameRelayDLCI</td>
</tr>
<tr>
<td>DiscoveredFrameRelayLMI</td>
<td>DiscoveredHBA</td>
<td>DiscoveredListeningPort</td>
</tr>
<tr>
<td>DiscoveredMFPart</td>
<td>DiscoveredMQDetail</td>
<td>DiscoveredMainframe</td>
</tr>
<tr>
<td>DiscoveredNetworkConnection</td>
<td>DiscoveredNetworkInterface</td>
<td>DiscoveredProcess (see Note below)</td>
</tr>
<tr>
<td>DiscoveredRegistryEntry</td>
<td>DiscoveredSNMPRow</td>
<td>DiscoveredService</td>
</tr>
<tr>
<td>DiscoveredSoftware</td>
<td>DiscoveredStorageSubsystem</td>
<td>DiscoveredSysplex</td>
</tr>
<tr>
<td>DiscoveredTapeDrive</td>
<td>DiscoveredTransaction</td>
<td>DiscoveredVirtualMachine</td>
</tr>
<tr>
<td>DiscoveredWBEMAssociatorsResult</td>
<td>DiscoveredWBEMInstance</td>
<td>DiscoveredWBEMQueryResult</td>
</tr>
<tr>
<td>DiscoveredWMI</td>
<td>IntegrationResultMember</td>
<td>SQLResultRow</td>
</tr>
<tr>
<td>vSphereIntegrationResult</td>
<td>DiscoveredCommandResult</td>
<td>DiscoveredFile</td>
</tr>
</tbody>
</table>
Simple Identity

Discovered Process nodes also contain one piece of information that is not Directly Discovered Data which is called a simple_identity. This is an inferred attribute value. It is set by identify tables defined in the Pattern Language (TPL), to quickly and simply identify known processes. The simple_identity attribute is only intended to be an initial hint about the purpose of a process; processes that are sufficiently important for complete identification have associated Software Instance nodes created in the inferred model.

Integration DDD nodes

Integration related DDD nodes are created when using Integration points to extract business contextual information from external data sources, or when performing deeper discovery of discovered databases by means of patterns and Configuring database credentials.

- Integration Point node
- Integration Result node
  - SQL Result Row node
- Provider Access node

Integration Point node

An Integration Point is a definition of how BMC Atrium Discovery should connect to and interact with an external data source.

There are two types of IntegrationPoint nodes, they are distinguished by the value of the is_dynamic attribute.

- If is_dynamic is false then the node represents an Integration Point entity. These are created through a pattern's definitions block whose type is sql_integration. These support querying SQL databases at well-known locations, for example asset databases.
- If is_dynamic is true then the node represents a Software Credential Group entity. These are created through a pattern's definitions block whose type is sql_discovery. These support querying SQL databases as they are discovered throughout the estate.

Integration Point Lifecycle

The following section describes the scenarios in which an Integration Point is created, updated or destroyed.
Creation/update
The creation and update of Integration Point nodes is done manually by the User using the BMC
Atrium Discovery UI, or as a result of uploading pattern modules containing appropriate
definitions blocks. See Integration points for more details.

Removal
Integration Points are removed manually by the user. When an Integration Point is destroyed its
child SQL Integration Connection and SQL Integration Query nodes are also destroyed.

Integration Point node attributes
The attributes and relationships of an Integration Point node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>The name of the integration point.</td>
</tr>
<tr>
<td>Description</td>
<td>description string</td>
<td>The description of what the integration point aims to achieve.</td>
</tr>
<tr>
<td>Provider Type</td>
<td>provider_type string</td>
<td>The type of the provider.</td>
</tr>
<tr>
<td>Dynamic Connections</td>
<td>is_dynamic boolean</td>
<td>Whether the integration point is for dynamic (endpoint supplied) or static (endpoint fixed)</td>
</tr>
</tbody>
</table>

Integration Point node relationships
The relationships of an Integration Point node are described in the following table.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>

Integration Result node
An Integration Result node is a record of the result of running SQL queries against a remote
database. It does not contain the actual data returned by the queries, this is stored in the related
SQL Result Row nodes.

An Integration Result node is related to either the Provider Access node for the endpoint at which
the database resides, in the case of database integration, or the Discovery Access node used for
the discovery of that database instance, in the case of database discovery.

Integration Result Lifecycle
The following section describes the scenarios in which an Integration Result is created, updated or
destroyed.
Creation
An Integration Result node is created whenever a pattern makes a new request to a SQL database.

Update
Integration Result nodes are never updated.

Removal
Integration Results are removed when the related Provider Access node or Discovery Access node is destroyed.

Integration Result node attributes
The attributes and relationships of an Integration Result node are described in the tables below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result Description</td>
<td>label string</td>
<td>A textual summary of the request made</td>
</tr>
<tr>
<td>Hash</td>
<td>hash string</td>
<td>A hash that uniquely identifies this request</td>
</tr>
<tr>
<td>Discovery duration</td>
<td>discovery_duration int</td>
<td>The length of time that the request took.</td>
</tr>
<tr>
<td>Request time</td>
<td>request_time date</td>
<td>When this request was made.</td>
</tr>
<tr>
<td>Failure reason</td>
<td>failure_reason date</td>
<td>If the request failed, this will have a textual explanation of why it failed.</td>
</tr>
<tr>
<td>Hard failure</td>
<td>hard_failure boolean</td>
<td>True if the result is classed as a hard failure.</td>
</tr>
<tr>
<td>Cancelled</td>
<td>cancelled boolean</td>
<td>True if command was cancelled.</td>
</tr>
<tr>
<td>Provider</td>
<td>provider string</td>
<td>The name of the provider that handled this request to an integration.</td>
</tr>
<tr>
<td>Connection Parameters</td>
<td>connection_xxx Any type</td>
<td>The value of a given connection parameter passed from the pattern.</td>
</tr>
<tr>
<td>Query Parameters</td>
<td>query_xxx Any type</td>
<td>The value of a given query parameter passed from the pattern.</td>
</tr>
</tbody>
</table>

Integration Result node relationships
The relationships of an Integration Result node are described in the following table.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQLIntegrationConnection</td>
<td>IntegrationResult: ConnectionUsage: IntegrationConnection: SQLIntegrationConnection</td>
<td>The connection that was used to produce this result</td>
</tr>
</tbody>
</table>
### SQL Result Row node

An SQL Result Row node is a record of a row of data returned when executing a SQL query against a remote database. Therefore any Integration Result node will have one or more SQL Result Rows related to them.

It is the data in the SQL Result Row nodes that is consumed by patterns.

#### SQL Result Row Lifecycle

The following section describes the scenarios in which an SQL Result Row is created, updated or destroyed.

**Creation**

SQL Result Row nodes are created when the related Integration Result node is created, as the result of running a SQL query.

**Update**

SQL Result Row nodes are never updated.

**Removal**

SQL Result Rows are removed when the related Integration Result node is destroyed.

### SQL Result Row node attributes
There are no pre-defined attributes on SQL Result Row nodes. The reason for this is that the attributes are entirely dynamic and depend on the queries that are being executed. If your SQL query results in many rows with a given number of columns, then each SQL Result Row node will have attributes for each of the columns, the value of which will be the value of the corresponding cell.

**SQL Result Row node relationships**

The relationships of an SQL Result Row node are described in the following table.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntegrationResult</td>
<td>SQLResultRow:</td>
<td>Associated integration result</td>
</tr>
<tr>
<td></td>
<td>Member:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>List:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>List:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IntegrationResult</td>
<td></td>
</tr>
</tbody>
</table>

**Provider Access node**

A Provider Access node represents the accesses to integrations (for example, a database integration or a query using the vSphere API) for a specific endpoint for a given **Discovery Run node**. It has relationships to each of the **Integration Result node** queries made by patterns to that database integration.

The endpoint attribute of a Provider Access node describes the IP v4 address at which that SQL database integration resides.

**Provider Access Lifecycle**

The following section describes the scenarios in which a Provider Access is created, updated or destroyed.

**Creation**

A Provider Access node is created when the first query is made to a database integration at that endpoint during the course of a Discovery Run. A Provider Access node is always created and stored, whether individual **Integration Result node** queries succeed or fail.

**Update**

The **endpoint** attribute on a Provider Access node is never updated. The **hard_failure** attribute can become true if a newly performed query represents a hard failure.

**Removal**

Provider Access nodes are removed when the Discovery Run node they are linked to is destroyed. See **Discovery Run Node** for more details. When a Provider Access node is destroyed, its related **Integration Result nodes** and their children are also destroyed.

**Provider Access node attributes**

The attributes and relationships of a Provider Access node are described in the tables below.
### Provider Access node relationships

The relationships of a Provider Access node are described in the following table.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Results</td>
<td>ProviderAccess: DiscoveryAccess: DiscoveryAccessResult: DiscoveryResult: IntegrationResult</td>
<td>The result of the access.</td>
</tr>
<tr>
<td>Discovery Run</td>
<td>ProviderAccess: Member: List: List: DiscoveryRun</td>
<td>Discovery Run.</td>
</tr>
<tr>
<td>Session Results</td>
<td>ProviderAccess: DiscoveryAccess: Metadata: Detail: SessionResult</td>
<td>Session details.</td>
</tr>
</tbody>
</table>

### Pattern Management nodes

This section describes the BMC Atrium Discovery Pattern Management nodes. The way in which BMC Atrium Discovery behaves is driven by patterns which represent the products that are identified and how they are identified. All of that information is stored in the model in the Pattern Management nodes.

<table>
<thead>
<tr>
<th>Provider Access Key</th>
<th>key string</th>
<th>provider_type/endpoint.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider Access Type</td>
<td>provider_type string</td>
<td>The type of the provider.</td>
</tr>
<tr>
<td>Range Prefix</td>
<td>range_prefix string</td>
<td>Distinguish overlapping address spaces.</td>
</tr>
<tr>
<td>Endpoint</td>
<td>endpoint string</td>
<td>The host that was accessed.</td>
</tr>
<tr>
<td>Dynamic Connections</td>
<td>is_dynamic boolean</td>
<td>Whether the integration point is for dynamic (endpoint supplied) or static (endpoint fixed).</td>
</tr>
<tr>
<td>Hard Failure</td>
<td>hard_failure boolean</td>
<td>True if there have been hard failures for this Provider Access type.</td>
</tr>
<tr>
<td>Retrieved by scanning appliance</td>
<td>is_consolidation boolean</td>
<td>If present, always have the value True and indicates that the data for this ProviderAccess originated from a scanning appliance before being consolidated on this one.</td>
</tr>
</tbody>
</table>
The default data model diagram, see The default data model, refers to the Pattern Management nodes as the 'Knowledge View'. Knowledge information is static information about software that exists in the world, rather than information about which instances of software are actually used in an environment. The majority of the knowledge information used in BMC Atrium Discovery comes from the Technology Knowledge Network (TKN); the rest is generated by customers to represent the in-house software they use. See Configipedia for more information.

- Pattern node describes the Pattern node kind.
- Pattern Module node describes the Pattern Module node kind.
- Knowledge Upload node describes the Knowledge Upload node kind.
- Rule Module node describes the Rule Module node kind.
- Pattern Configuration nodes describes the Pattern Configuration node kind.
- Pattern Definitions node describes the Pattern Definitions node kind.
- Pattern Define nodes describes the Pattern Define node kind.

Pattern node

The Pattern node is the node that represents an individual pattern, defined in the Pattern Language. For each pattern in a TPL file there is a Pattern node which contains meta-data about the pattern, such as its name, description and the products that it is responsible for identifying.

Patterns have relationships to all of the nodes that they are responsible for maintaining so that the nodes can be related back to where they originated. See Configipedia for more information about BMC Atrium Discovery patterns.

Pattern node attributes

The attributes of a Pattern node are described below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Name</td>
<td>Pattern name.</td>
</tr>
<tr>
<td></td>
<td>name string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pattern Version</td>
<td>Pattern version.</td>
</tr>
<tr>
<td></td>
<td>version string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tags</td>
<td>Tags.</td>
</tr>
<tr>
<td></td>
<td>tags list:string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>Description.</td>
</tr>
<tr>
<td></td>
<td>description string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Trigger Kind</td>
<td>Trigger node or relationship kind.</td>
</tr>
<tr>
<td></td>
<td>_trigger_kind string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Triggers On Relationship</td>
<td>Flag indicating if trigger kind is a node (False) or relationship (True).</td>
</tr>
<tr>
<td></td>
<td>_trigger_is_rel boolean</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pattern Active</td>
<td>Flag indicating if Pattern is active.</td>
</tr>
<tr>
<td></td>
<td>__active boolean</td>
<td></td>
</tr>
<tr>
<td>UI Name</td>
<td>Attribute Name and Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>Products</td>
<td>products list:string</td>
<td>Products identified by the pattern.</td>
</tr>
<tr>
<td>Product Synonyms</td>
<td>product_synonyms list:string</td>
<td>Synonyms for products identified by the pattern.</td>
</tr>
<tr>
<td>Known Versions</td>
<td>known_versions list:string</td>
<td>Known versions of products identified by the pattern.</td>
</tr>
<tr>
<td>Publishers</td>
<td>publishers list:string</td>
<td>Publishers of products identified by the pattern.</td>
</tr>
<tr>
<td>Publisher Synonyms</td>
<td>publisher_synonyms list:string</td>
<td>Synonyms for publishers of products identified by the pattern.</td>
</tr>
<tr>
<td>OEM Publishers</td>
<td>oem_publishers list:string</td>
<td>OEM publishers of products identified by the pattern.</td>
</tr>
<tr>
<td>Prior Publishers</td>
<td>prior_publishers list:string</td>
<td>Previous publishers of products identified by the pattern.</td>
</tr>
<tr>
<td>Product Families</td>
<td>product_families list:string</td>
<td>Product families of products identified by the pattern.</td>
</tr>
<tr>
<td>Product Editions</td>
<td>product_editions list:string</td>
<td>Editions of products identified by the pattern.</td>
</tr>
<tr>
<td>Categories</td>
<td>categories list:string</td>
<td>Categories of products identified by the pattern.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>cdm_software_server_type list:int</td>
<td>Software server type in the Common Data Model.</td>
</tr>
<tr>
<td>URLs</td>
<td>urls list:string</td>
<td>URLs relating to the pattern.</td>
</tr>
<tr>
<td>TKN Identity</td>
<td>tkn_identity list:string</td>
<td>Identity in the Technology Knowledge Network.</td>
</tr>
<tr>
<td>Additional Attributes</td>
<td>additional_attributes list:string</td>
<td>Additional attributes that can be set on nodes maintained by the pattern.</td>
</tr>
</tbody>
</table>

**Pattern node relationships**

The relationships of a Pattern node are described below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintained Software Instances</td>
<td>Pattern: Pattern: Maintainer: Element: SoftwareInstance</td>
<td>Software Instances maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Software Components</td>
<td>Pattern: Pattern: Maintainer: Element: SoftwareComponent</td>
<td>Software Components maintained by this pattern.</td>
</tr>
<tr>
<td>UI name</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Maintained Business Application Instances</td>
<td>Pattern: Pattern: Maintainer: Element: BusinessApplicationInstance</td>
<td>Business Application Instances maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Host Container</td>
<td>Pattern: Pattern: Maintainer: Element: HostContainer</td>
<td>Host Container maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Detail</td>
<td>Pattern: Pattern: Maintainer: Element: Detail</td>
<td>Detail maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Database Detail</td>
<td>Pattern: Pattern: Maintainer: Element: DatabaseDetail</td>
<td>Database Detail maintained by this pattern.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Pattern: Pattern: Maintainer: Element: SupportDetail</td>
<td>Support Detail maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Generic Element</td>
<td>Pattern: Pattern: Maintainer: Element: GenericElement</td>
<td>Generic Element maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Collection</td>
<td>Pattern: Pattern: Maintainer: Element: Collection</td>
<td>Collection maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Storage Collection</td>
<td>Pattern: Pattern: Maintainer: Element: StorageCollection</td>
<td>Storage Collection maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained Storage</td>
<td>Pattern: Pattern: Maintainer: Element: Storage</td>
<td>Storage maintained by this pattern.</td>
</tr>
<tr>
<td>Maintained File</td>
<td>Pattern: Pattern: Maintainer: Element: File</td>
<td>File maintained by this pattern.</td>
</tr>
<tr>
<td>Requires</td>
<td>Pattern: Dependant: Dependency: DependedUpon: Pattern</td>
<td>Required patterns.</td>
</tr>
<tr>
<td>Required By</td>
<td>Pattern: DependedUpon: Dependency: Dependant: Pattern</td>
<td>Patterns that depend on this one.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>Pattern: Contributor: Inference: InferredElement: SoftwareInstance</td>
<td>Software instance whose attributes have been partly or wholly determined from this pattern.</td>
</tr>
<tr>
<td>Deprecating pattern</td>
<td>Pattern: Old: Deprecated: New: Pattern</td>
<td>Pattern that deprecated this one.</td>
</tr>
<tr>
<td>Deprecated pattern</td>
<td>Pattern: New: Deprecated: Old: Pattern</td>
<td>Pattern that has been deprecated by this one.</td>
</tr>
<tr>
<td>Pattern Errors</td>
<td>Pattern: PatternWithError: Error: Error: ECAError</td>
<td>Errors generated by this pattern.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Pattern resource.</td>
</tr>
</tbody>
</table>
Patterns support having configuration that allows the behavior of the pattern to be modified without needing to learn and edit TPL. This data is stored in one or more Pattern Configuration nodes that are related to a Pattern Module node. As a pattern's configuration cannot be pre-determined, the default list of attributes for a Pattern Configuration node is limited to generics.

Pattern Configuration node lifecycle
The following section describes the scenarios in which a Pattern Configuration node is created, updated or removed.

Creation
Pattern Configuration nodes are created as a result of activating a Pattern Module that contains a configuration block. As such the creation of Pattern Configuration nodes is driven by the management of Patterns. See Knowledge management for details.

Update
The update of a Pattern Configuration node is driven by the User modifying the attributes in the UI.

Removal
Pattern Configuration nodes are removed when their associated Pattern Modules are destroyed.

Pattern Configuration node attributes
The attributes of a Pattern Configuration node are described below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Name.</td>
</tr>
<tr>
<td>Pattern Configuration Version</td>
<td>version int</td>
<td>Version.</td>
</tr>
<tr>
<td>Pattern Configuration Description</td>
<td>description string</td>
<td>Description.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__active boolean</td>
<td>Flag indicating if configuration is active.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__config_values list:string</td>
<td>List of name for configuration values.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__changed_values int</td>
<td>Number of non-default values.</td>
</tr>
</tbody>
</table>
Pattern Configuration Relationships
The relationships of a Pattern Configuration node are described below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PatternModule</td>
<td>PatternConfiguration: PatternModuleContainment: PatternModule: PatternModule</td>
<td>Pattern module</td>
</tr>
<tr>
<td>Deprecated</td>
<td>PatternConfiguration: Old: Deprecated: New: PatternConfiguration</td>
<td>PatternConfiguration that deprecated this one.</td>
</tr>
<tr>
<td>Deprecated</td>
<td>PatternConfiguration: New: Deprecated: Old: PatternConfiguration</td>
<td>PatternConfiguration that has been deprecated by this one.</td>
</tr>
</tbody>
</table>

Pattern Define nodes
A Pattern Define node is created for every `define` block in the `definitions` block for a Pattern Module node. See Pattern Management TPL for more details on `define` blocks.

Pattern Define node lifecycle
The following section describes the scenarios in which a Pattern Define node is created, updated or removed.

Creation
Pattern Define nodes are created as a result of activating a Pattern Module that contains a `definitions` block that has one or more `define` blocks. As such the creation of Pattern Define nodes is driven by the management of Patterns. See Knowledge management for details.

Update
Pattern Define nodes are never updated as they are defined in Pattern Modules and the act of `editing` a Pattern Module creates a new instance.

Removal
Pattern Define nodes are removed when their associated Pattern Modules are destroyed.

Pattern Define node attributes
The attributes of a Pattern Define node are described below.
<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Pattern definitions name.</td>
</tr>
<tr>
<td>Pattern Definitions</td>
<td>description string</td>
<td>Pattern definitions description.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__config_values list:string</td>
<td>List of name for definition values.</td>
</tr>
</tbody>
</table>

Pattern Define node relationships

The relationships of a Pattern Define node are described below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>

Pattern Definitions node

A Pattern Definitions node is a container for all the Pattern Define nodes in a Pattern Module. Currently Pattern Definitions are only used in the context of SQL Discovery and Integration.

Pattern Definitions node lifecycle

The following section describes the scenarios in which a Pattern Definitions node is created, updated or removed.

Creation

Pattern Definitions nodes are created as a result of activating a Pattern Module that contains a definitions block. As such the creation of Pattern Definitions nodes is driven by the management of Patterns. See Knowledge management for details.

Update

Pattern Definitions nodes are never updated as they are defined in Pattern Modules and the act of editing a Pattern Module creates a new instance.

Removal

Pattern Definitions nodes are removed when their associated Pattern Modules are destroyed.
Pattern Definitions node attributes

The attributes of a Pattern Definitions node are described below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>name string</td>
<td>Pattern definitions name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern Definitions Version</td>
<td>version int</td>
<td>Pattern definitions version.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern Definitions Description</td>
<td>description string</td>
<td>Pattern definitions description.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern Definitions Type</td>
<td>type string</td>
<td>Pattern definitions type.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__active boolean</td>
<td>Flag indicating if definitions are active.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__config_values list:string</td>
<td>List of name for definition values.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>__define_values list:string</td>
<td>Order of define blocks in this definitions block.</td>
</tr>
</tbody>
</table>

Pattern Definitions Relationships

The relationships of a Pattern Definitions node are described below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PatternModule</td>
<td>PatternDefinitions: PatternDefinitions: PatternModuleContainment: PatternModule: PatternModule</td>
<td>Pattern module</td>
</tr>
<tr>
<td></td>
<td>PatternDefinitions: Old: Deprecated: New: PatternDefinitions</td>
<td>PatternDefinitions that deprecated this one</td>
</tr>
<tr>
<td></td>
<td>PatternDefinitions: New: Deprecated: Old: PatternDefinitions</td>
<td>PatternDefinitions that has been deprecated by this one</td>
</tr>
<tr>
<td></td>
<td>PatternDefinitions: PatternDefinitions: PatternModuleContainment: PatternDefine: PatternDefine</td>
<td>Pattern definitions define functions</td>
</tr>
<tr>
<td>IntegrationPoint</td>
<td></td>
<td>Implements resource</td>
</tr>
<tr>
<td>Destination</td>
<td>Relationship</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PatternDefinitions:</td>
<td>IntegrationImplementation:</td>
<td>IntegrationPoint: IntegrationPoint</td>
</tr>
</tbody>
</table>

**Pattern Module node**

A Pattern Module node represents a TPL file. Each file in the Pattern Language is a pattern module and therefore has a Pattern Module node. The Pattern Module node contains meta-data about the module, such as its name and description. This node also contains the entire text of the TPL file in the content attribute.

**Pattern Module node attributes**

The attributes of a Pattern Module node are described below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>name string</td>
<td>Module name.</td>
</tr>
<tr>
<td>Module Type</td>
<td>type string</td>
<td>Module type.</td>
</tr>
<tr>
<td>Content</td>
<td>content string</td>
<td>Module text. Contains the entire text of the TPL file.</td>
</tr>
<tr>
<td>Source MD5 Sum</td>
<td>source_md5sum string</td>
<td>Source MD5 sum.</td>
</tr>
<tr>
<td>Compiled syncmapping MD5 Sum</td>
<td>syncmapping_md5sum string</td>
<td>Compiled syncmapping MD5 sum.</td>
</tr>
<tr>
<td>Submitting User</td>
<td>submitting_user string</td>
<td>Submitting user.</td>
</tr>
<tr>
<td>Submission Date</td>
<td>submission_date date</td>
<td>Date and time of submission.</td>
</tr>
<tr>
<td>Active</td>
<td>active boolean</td>
<td>True if module is active.</td>
</tr>
<tr>
<td>Activation Date</td>
<td>compiled_date date</td>
<td>Date and time of activation.</td>
</tr>
<tr>
<td>Upgrade messages</td>
<td>upgrade_messages list:string</td>
<td>Messages generated during upgrade recompilation of this pattern module.</td>
</tr>
<tr>
<td>Not displayed in UI</td>
<td>tree_path list:string</td>
<td>Path in Pattern Module tree.</td>
</tr>
</tbody>
</table>

⚠️ In BMC Atrium Discovery v10.0, the following attribute changes occurred:
enabled has been renamed to active.
flagged_by_upgrade and compile_messages attributes were replaced with
the upgrade_messages attribute.

Pattern Module node relationships
The relationships of a Pattern Module node are described below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depends On</td>
<td>PatternModule: Dependant: Dependency: DependedUpon: PatternModule</td>
<td>Pattern modules that are depended upon.</td>
</tr>
<tr>
<td>Depended Upon By</td>
<td>PatternModule: DependedUpon: Dependency: Dependant: PatternModule</td>
<td>Pattern module that are dependants.</td>
</tr>
<tr>
<td>Generated From</td>
<td>PatternModule: Output: CodeGeneration: Input: Group</td>
<td>Group which generated this pattern module.</td>
</tr>
<tr>
<td>Generated From</td>
<td>PatternModule: Output: CodeGeneration: Input: UIPatternDefinition</td>
<td>Pattern definition which generated this pattern module.</td>
</tr>
</tbody>
</table>

⚠️ In BMC Atrium Discovery v10.0, the Pattern Module node does not include Pattern Package relationship.

Knowledge Upload node
A Knowledge Upload node represents a set of pattern modules uploaded from a single file. The file can be a single module or multiple modules zipped together.

Knowledge Upload node attributes
The attributes of a Knowledge Upload node are described below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upload name string</td>
<td>Upload name.</td>
</tr>
</tbody>
</table>
### Knowledge Upload node relationships

The relationships of a Knowledge Upload node are described below.

<table>
<thead>
<tr>
<th>UI name</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
</table>

### Rule Module node

Patterns are compiled into one or more rules which execute in the Reasoning Engine. For implementation reasons, BMC Atrium Discovery represents these rules as nodes in the datastore. The Rule Module node is for BMC internal use only.

### Rule Module node attributes

The attributes of a Rule Module node are described below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type or Relationship Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule Module Name name string</td>
<td>Name.</td>
</tr>
<tr>
<td>Version version int</td>
<td>Version.</td>
</tr>
<tr>
<td>Description description string</td>
<td>Description.</td>
</tr>
<tr>
<td>Source MD5 Checksum source_md5sum string</td>
<td>Source MD5 checksum.</td>
</tr>
<tr>
<td>Compiled MD5 Checksum compiled_md5sum string</td>
<td>Compiled MD5 checksum.</td>
</tr>
</tbody>
</table>
Rule Module Relationships

The relationships of a Rule Module node are described below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depends On</td>
<td>Rule Module:Dependant:Dependency:DependedUpon:RuleModule</td>
<td>Rule modules that are required.</td>
</tr>
<tr>
<td>Depended Upon By</td>
<td>Rule Module:DependedUpon:Dependency:Dependant:RuleModule</td>
<td>Rule modules that are dependants.</td>
</tr>
</tbody>
</table>

Root Node Key Info node

Key information about root nodes, imported from another instance of BMC Atrium Discovery. The Root Node Key Info node is for BMC internal use only.

Root Node Key Info node attributes

The attributes of a Root Node Key Info node are described below.

<table>
<thead>
<tr>
<th>UI Name Attribute Name and Type or Relationship Target</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Node Key key string</td>
<td>Unique root node key.</td>
</tr>
<tr>
<td>Root Node Kind root_node_kind string</td>
<td>Kind of root node.</td>
</tr>
<tr>
<td>Hostname hostname string</td>
<td>Host name.</td>
</tr>
<tr>
<td>System Name sysname string</td>
<td>System name.</td>
</tr>
</tbody>
</table>

Conjecture nodes

This section describes the BMC Atrium Discovery Conjecture nodes. Conjectured information is derived from observed evidence using specialized algorithms. For example, Host Groups are Conjecture nodes. There is no attribute on individual hosts that states that they belong in a particular group. However, using observed network communication between hosts BMC Atrium Discovery can assign hosts to logical groups. See Automatic Group node for more information.

- Automatic Group node describes the node kind that is used to model host groups.
**Automatic Group node**

An Automatic Group is the term used for a number of hosts that are grouped according to observed network communications using the Automatic grouping feature. The sole purpose of an Automatic Group node is to build the visualizations that are part of the feature. Automatic Group nodes are only created or destroyed if the Enable Automatic Grouping setting is enabled on the Discovery Configuration page.

Automatic Group nodes are stored in the Conjecture partition.

**Automatic Group Lifecycle**

The following section describes the scenarios in which an Automatic Group is created, updated or removed.

**Creation**

Automatic Group nodes are created at the end of every discovery run, or when the Update button in the automatic grouping UI is clicked.

**Update**

Automatic Group nodes are not updated.

**Removal**

Automatic Group nodes are destroyed at the end of every discovery run, or when the Update button in the automatic grouping UI is clicked. New Automatic Group nodes are then created taking into account any changes to hosts and network communications.

**Automatic Group node attributes**

The attributes and relationships of an Automatic Group node are described in the table below.

<table>
<thead>
<tr>
<th>UI Name</th>
<th>Attribute Name and Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Details</td>
<td>name string</td>
<td>The name of the Automatic Group.</td>
</tr>
<tr>
<td></td>
<td>type string</td>
<td>The type of the Automatic Group.</td>
</tr>
</tbody>
</table>

**Automatic Group node relationships**

The relationships of an Automatic Group node are described in the table below.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Relationship</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Members</td>
<td>AutomaticGroup: List: List: Member: Host</td>
<td>Hosts which are members of this Automatic Group.</td>
</tr>
</tbody>
</table>
The Pattern Language TPL

The Pattern Language, TPL, is used to describe applications, products and other real-world entities that have been modeled in BMC Atrium Discovery. The core purpose of TPL is to make it easy to describe the most common structures to be modeled, while remaining powerful enough to describe the vast majority of modeling situations.

The syntax and semantics of the pattern language are described in this reference guide. If you are new to TPL, start with the Introduction to Collaborative Application Mapping.

- What's new in TPL
- TPL recommendations
- TPL file structure
- Import
- Metadata
- Pattern Overview
- Pattern Configuration
- Static Tables
- Enumerations
- Identify
- Definitions
- Node names types and keys
- Syncmapping block
- TPL Keywords
- TPL Grammar
- String Escape Characters
- Using TPL to enrich discovered data
- VMware ESX and ESXi hosts discovery using custom patterns
- Displaying custom attributes

What's new in TPL

New in TPL 1.10/BMC Atrium Discovery 10.2

TPL version 1.10 is available with BMC Atrium Discovery 10.2. It is a minor update to TPL version 1.9.

The following functions have been added:

- binary.toValue
- overrides in syncmapping block
- Impacted and ImpactWeight in syncmapping block
New in TPL 1.9/BMC Atrium Discovery 10.1
TPL version 1.9 is available with BMC Atrium Discovery 10.1. It is a minor update to TPL version 1.8.

The following functions have been added:

- `discovery.wbemEnumInstances`
- `discovery.getNames`
- `number.toChar`
- `binary.toIPv4z`
- `binary.toIPv6z`

New in TPL 1.8/BMC Atrium Discovery 10.0
TPL version 1.8 is available with BMC Atrium Discovery 10.0. It is a minor update to TPL version 1.7.

- Add `model.addDisplayAttribute` and `model.removeDisplayAttribute` to modify `_tw_meta_data_attrs`
- Extend `discovery.snmpGet` and `discovery.snmpGetTable` to allow caller to distinguish binary data
- Add `binary.toIPv4`, `binary.toIPv6`, `binary.toMACAddress`, `binary.toWWN` and `binary.toHexString` to convert binary data to text.

New in TPL 1.7/BMC Atrium Discovery 9.0
TPL version 1.7 is available with BMC Atrium Discovery 9.0. It is a minor update to TPL version 1.6.

TPL 1.7 adds the following new features:

- New `text.ordinal` function. See Text functions.

New in TPL 1.6/BMC Atrium Discovery 8.3
TPL version 1.6 is available with BMC Atrium Discovery 8.3. It is a minor update to TPL version 1.5.

TPL 1.6 adds the following new features:

- Removal groups. See the following functions:
  - `model.setRemovalGroup`
  - `model.anchorRemovalGroup`
  - `model.suppressRemovalGroup`
A new 

**void** keyword is added. Assigning this to an attribute in the model deletes the attributes. The statement 

```plaintext
node['attribute_name'] := void;
```

is the equivalent of 

```plaintext
del node['attribute_name']
```

**String and list slicing**

**New in TPL 1.5/BMC Atrium Discovery 8.2**

TPL version 1.5 is available with BMC Atrium Discovery 8.2. It is a minor update to TPL version 1.4.

TPL 1.5 adds the following new features:

- User defined functions. See [User defined functions](#).
- You can now reference constants defined in a pattern from other patterns. See [Constants](#).
- Version restrictions are now enforced according to the version specified in the pattern. Prior to TPL 1.5 the TPL version specified at the top of a pattern file was only checked against the supported version in BMC Atrium Discovery. Now functionality defined in a specific TPL version is available if that (or a later) version is specified in the pattern file.

**New in TPL 1.4/BMC Atrium Discovery 8.1**

TPL version 1.4 is available with BMC Atrium Discovery 8.1. It is a minor update to TPL version 1.3.

TPL 1.4 adds the following new features:

- Enumerations, this is described in [Enumerations](#).
- New `discovery.listRegistry` function. See [Discovery functions](#).

**New in TPL 1.3/Tideway Foundation 7.3**

TPL version 1.3 is available with Tideway Foundation 7.3. It is a minor update to TPL version 1.2.

TPL 1.3 adds a number of new features:

- New `discovery.listDirectory` function. See [Discovery functions](#).

**New in TPL 1.2/Tideway Foundation 7.2**

TPL version 1.2 is available with Tideway Foundation 7.2. It is a minor update to TPL version 1.1.

TPL 1.2 adds a number of new features:

- Pattern configuration blocks allow end users to modify parameters without editing pattern text. See [Pattern Configuration](#).
- New definitions blocks support SQL Discovery (now Integration Points and Configuring database credentials). See [Definitions](#).
- Model maintenance is made easier with the addition of `unique relationship` functions. See [Unique relationships](#).
• Lookup tables can be created and manipulated inside patterns. See Table functions.
• New functions to manipulate dates and times are available. See Time functions.

New in TPL 1.1/Tideway Foundation 7.1
TPL version 1.1 is available with Tideway Foundation 7.1. It is a minor update to TPL version 1.0.

It has the following new features:

• New confirmed trigger kind. See Trigger kind.
• New text.strip, text.leftStrip and text.rightStrip functions. See Text functions.
• New mail.send function. See Email functions.
• New discovery.snmpGet and discovery.snmpGetTable functions. See Discovery functions.

TPL recommendations
The core purpose of TPL is to make it easy to describe the most common structures to be modeled, while remaining powerful enough to describe the vast majority of modeling situations. While the power of TPL enables you to describe the vast majority of modeling situations, it does not prevent you creating patterns that cause unwanted, or unexpected behavior.

This page details recommendations for using TPL. Following these recommendation will prevent unwanted behavior.

Never modify system created attributes
When writing custom TPL you should never modify system created attributes. Rather, you should add new attributes using _tw_meta_data_attrs as described in adding custom attributes.

System created attributes are those present in the default taxonomy. You can see whether your taxonomy is customized by checking for additional custom files on the Files tab of the Taxonomy page.

Use related.host rather than model.host
The related.host function should be used rather than the deprecated model.host function. The model.host function, with a lower-case "h" has an unfortunate clash with model.Host with an upper-case "H", which creates or updates an existing Host node.

Avoid modifying TKU patterns
When writing TPL patterns it is recommended that you do not modify TKU patterns. If you do modify a TKU pattern, it will be overwritten when you install a later TKU. When writing a pattern to override a TKU pattern, you should use the overrides statement in the pattern declaration.
Avoid custom pattern module name collision

Custom pattern does not trigger if the custom module name collides with a TKU module name. There is no error or warning; it just silently fails when scanning. To avoid this you can add a name, for example the company name, ACME_, at the beginning of any custom pattern name, custom module name, and custom attribute name.

TPL file structure

Typographical conventions

The description of the Pattern Language uses a number of typographical conventions to distinguish items, as follows:

- **fixed pitch font** — TPL code fragments.
- **italic fixed pitch** — Items to be replaced in code fragments.
- **bold fixed pitch** — TPL keywords.
- **[items in square brackets]** — optional items.

Pattern language files are simple flat text files, using UTF-8 character encoding.

TPL is a case sensitive language.

Comments are prefixed with two slash characters, as in C++ and Java:

```plaintext
// This is a comment
```

The first non-blank, non-comment line must consist of a declaration of the form:

```plaintext
tpl 1.n module module.name;
```

where `module.name` is a dot-separated name for the module in the module hierarchy. The `1.n` specifies the version of TPL the module conforms to. The current version of TPL is 1.3, corresponding to Foundation 7.3. This document highlights which version of TPL each feature is available in. Patterns can fail to activate if they use features that are not available in the version declared in the module.

Following the module declaration, there can be any number of the following kinds of declaration, in any order that satisfies references between declarations:

- **Import** statements are used to import referenced declarations from other files.
- **Pattern** declarations define patterns, which are responsible for identifying entities and building the model.
- **Table** declarations are used to create look-up tables to be used in patterns.
- **Identify** table declarations define active look-up tables that automatically set identifying attributes on matching nodes.
- **Configuration** declarations define end-user editable configuration items.
• **Definitions** blocks are used to specify integration queries.

The statements and declarations are described in detail in the following sections.

**Common declarations**

A number of kinds of declarations can occur throughout pattern files.

**Strings**

Literal strings can be declared in four ways:

- Started with a single quote character, terminated by an unescaped single quote; cannot contain newline characters.
- Started with a double quote character, terminated by an unescaped double quote; cannot contain newline characters.
- Started with three single quote characters, terminated by three unescaped single quotes; can include newline characters.
- Started with three double quote characters, terminated by three unescaped double quotes; can include newline characters.

Triple-quoted strings are often used for multi-line string literals, especially for description strings in declarations.

The escape character for strings is the backslash "\". A backslash preceding a quote prevents it from terminating a string.

Strings can be "qualified" by prefixing them with a single identifier. The supported identifiers are:

- No qualifier — characters prefixed with backslashes are converted to special characters using the same rules as Python unicode strings, as shown in String Escape Characters. String interpolation using $% characters occurs, as described in section string interpolation.

- raw — backslashes do not create special characters, and string interpolation does not occur. Backslashes preceding a closing quote character still prevent closure of the string, but the backslash remains part of the string. i.e. in the string: *raw 'test\'quote'* both the backslash and the quote appear in the string value.

- regex — indicates that the string value is a regular expression. The contents are handled in the same way as raw strings.

- windows_cmd — the string contains the name of a Windows executable to be identified in a process command line. It turns into a suitable regular expression for identifying the process, formed by prefixing the given string with "(?i)\b" and suffixing it with "\.exe$".

- unix_cmd — the string contains the name of a UNIX executable to be identified in a process command line. It turns into a suitable regular expression for identifying the process, formed by prefixing the given string with "\b" and suffixing it with "$".
Numbers
Literal numbers are integers. Integers are in base 10 unless prefixed with $0x$ in which case they are hexadecimal. Integers are signed 64 bit values.

Booleans
Literal booleans are true and false.

None
The value none represents the absence of any other value.

void
Assigning this to an attribute in the model deletes the attributes. The statement node ['attribute_name'] := void; is the equivalent of del node ['attribute_name']

Identifiers
Identifiers used for names of variables, patterns, tags, and so on are comprised of only ASCII characters. They start with a letter or underscore character and contain letters, underscores and numbers. Names starting with two underscore characters are reserved for the system.

If it is necessary to declare an identifier with a name that clashes with a keyword, the keyword can be used prefixed with a $ character.

Tags
Many declarations contain tags, used to classify them with arbitrary names. Tags are useful when searching for declarations. Tags have the format:

    tags tag_value [, tag_value, ... ];

Tag values must be valid identifiers. No meaning is attached to tags by the pattern language.

Data types
All data values in TPL have a type, such as text strings and integers. Patterns must be careful to be consistent in the use of types. It is an execution-time error to attempt to add an integer to a text string, for example.

When attributes are set on datastore nodes and relationships, the attributes are set with the types of the supplied values, regardless of any type definitions in the taxonomy.

The types supported in TPL are: text string, integer, time, list and table. Each list or table member has its own type; usually each member has the same type, but lists and tables are permitted to have mixed-type members. List is a valid list member type, so lists can be arbitrarily nested.

Import
Simple import statements take the following form:
```
from module import name version;
```

This declaration makes the named declaration in the specified module available. Module names are dot-separated sequences of names. All imports are absolute, not relative to the importing module's name. For example, the declaration

```
from a.b.c import Example 1.0;
```

finds a module named "a.b.c" and imports the item (pattern, for example) with the name "Example". Multiple named declarations can be accessed in one statement by separating them with commas:

```
from a.b.c import Example 1.0, Another 2.3;
```

Imported names must not clash with other names declared or imported in the same file. To avoid clashes, declarations can be renamed on import using an `as` clause:

```
from one import Example 1.0; from two import Example 2.3 as ExampleTwo;
```

When importing, it is not immediately an error to import a name that is not defined in the corresponding module, or to import from a non-existent module. It only becomes an error when the imported name is used in a context that requires it exists. In particular, when a pattern overrides another pattern (see the section on `overrides`), it is not an error if the overridden pattern does not exist. For all other uses of an imported name, it is an error if the import fails.

Imported names are not available for further import. So, for example, if module B imports X from module A, module C cannot import X from B - it must import it directly from A.

**Version compatibility**

If there is an updated version of the selected object (for example, pattern or definition), the system will check your module containing the import to ensure that the minimum requirements are still met. For example, if you import version 1.1, this will allow the imported object to be 1.1, 1.2, and so forth, but NOT allow versions 1.0, 2.0, 2.1, 3.0, and so forth. The major version must be the same and the minor version must be at least the same as the version specified. If the version is not supported, the activation fails.

Most of the time, you will not need to update the import version. It might be necessary only if either there is an incompatible change in the imported pattern (represented by a major version change), or if you modify your pattern so that is relies on additional behaviour of the updated pattern. See **Pattern overview** for more information about how patterns are versioned.

**Recommendation**
Patterns contain tags that have information about the most recent Technology Knowledge Update (TKU) that updated the pattern (for example, TKU_2010_04_01). Therefore, you can perform a search for a specific TKU version to find patterns that were last changed in a specific version of TKU. Typing that value in the search box displays a list of patterns (and the containing pattern modules).

**Version numbers**

All top-level declarations are specified with a version number in major.minor format. When importing declarations, the version number must be specified. It is an error if the declaration imported is not compatible with the version specified in the import statement. The versions are compatible if the major number is identical and the minor number is at least as large as the requested version. So, for example, in

```
from test import Example 2.3;
```

the "Example" declaration from the "test" module is compatible if it has version 2.3 or 2.4, but not if it has version 2.2, 1.8 or 3.0.

**Circular imports**

Circular imports are not permitted. It is an error for file B to import from file A when file A is still being processed.

**Metadata**

Metadata can be provided for a number of declarations, describing the purpose of the declarations. Metadata attributes can be defined for the pattern module as a whole, and for individual patterns and identify tables in the module. Metadata declarations take the form:

```
metadata name := value1, value2, ...; ...end metadata;
```

When pattern modules are loaded into BMC Atrium Discovery, the metadata attributes are stored along with the pattern modules in the data store, allowing them to be searched for and used in reports. There is not a fixed set of metadata attributes - any attributes can be set as appropriate for the pattern. However, the following attributes are recommended for all patterns and pattern modules:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Expected?</th>
</tr>
</thead>
<tbody>
<tr>
<td>products</td>
<td>The names of the product(s) identified by the pattern</td>
<td>expected</td>
</tr>
<tr>
<td>product_synonyms</td>
<td>Other names for the product(s)</td>
<td>optional</td>
</tr>
<tr>
<td>known_versions</td>
<td>Versions of the product(s) that are known to exist</td>
<td>optional</td>
</tr>
<tr>
<td>publishers</td>
<td>The publisher(s) or the product(s) at the time of pattern creation</td>
<td>expected</td>
</tr>
<tr>
<td>publisher_synonyms</td>
<td>Other names the publishers are known by</td>
<td>optional</td>
</tr>
<tr>
<td>oem_publishers</td>
<td>Any publishers that OEM the product(s)</td>
<td>optional</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Expected?</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>prior_publishers</td>
<td>Publishers previously related to the product(s), in case the product or publisher has been bought</td>
<td>optional</td>
</tr>
<tr>
<td>product_families</td>
<td>Publisher specific product families the product(s) belong to</td>
<td>optional</td>
</tr>
<tr>
<td>product_editions</td>
<td>If the product is available in different editions, which ones the pattern covers</td>
<td>optional</td>
</tr>
<tr>
<td>categories</td>
<td>The software categories the product is in</td>
<td>expected</td>
</tr>
<tr>
<td>urls</td>
<td>Web URLs of relevance</td>
<td>optional</td>
</tr>
<tr>
<td>tkn_identity</td>
<td>For patterns from the Technology Knowledge Network, the identity in the TKN database</td>
<td>expected</td>
</tr>
<tr>
<td>tree_path</td>
<td>Position of the pattern module within the tree on the Knowledge management page</td>
<td>expected</td>
</tr>
</tbody>
</table>

All of the attributes are lists of strings (in TPL 1.4, numbers are also permitted) since it is possible for one single pattern or pattern module to be relevant to a number of different products. Attributes are tagged as "expected" if well-written patterns would be expected to have them. All patterns available through TKN will have all of the expected attributes.

**Pattern Overview**

Patterns are the main declarations in the pattern language. They are used to describe the structures of applications, products and other modeled entities, and encapsulate rules about how to identify them. The following sections provide descriptions of the various features of patterns:

- Overview
- Constants
- Triggers
- Body
- Removal

**Pattern declarations**

Pattern declarations have the form:

```
pattern name version
description
[metadata
metadata_entries
end metadata;]
overview
overview_entries
end overview;
[constants
constant_definitions
end constants;]
triggers
```
trigger_definitions
end triggers;
body
body_details
end body;
[removal
removal_condition
removal_details
end removal;]
end pattern;

The name, version and description are mandatory, for example:

pattern Xvnc 1.0
"""
X VNC server pattern.
Creates one SoftwareInstance per running VNC server.
"""
...

Pattern version numbers
It is very important that pattern version numbers are correctly specified, so as to avoid unexpected
results in the model. Whenever a pattern is modified, its version number should be incremented. If
the modification does not change the structure of any data constructed by the pattern, or only non-
standard node attributes are changed, the pattern's minorversion number should be incremented;
if the structure of the model constructed is changed, the majorversion should be incremented. In
particular, if the key attributes of any maintained nodes are changed, the pattern's major version
must be incremented.

When a new version of a pattern is uploaded into BMC Atrium Discovery, nodes maintained by the
previous version can be affected. On minor pattern version updates, the new pattern becomes the
maintainer for the nodes. On a major version update, nodes maintained by the previous version
continue to be maintained by the previous pattern; when the older pattern is removed, the nodes it
is maintaining are destroyed.

Overview section
The overview is required. It contains information about the pattern and the entities it creates. It
must contain a tags entry, and can have optional overrides and requires entries.

Tags
tags tag_name [ , tag ... ];

Declares tags relating to the pattern. At least one tag must be present. See the section on tags.
Overrides

overrides pattern_name [ , pattern_name ... ];

Indicates that the pattern overrides one or more other patterns. In that case, the other patterns are disabled.

The named pattern must be previously declared in the file, either with a pattern declaration, or via an import. It makes little sense to override a pattern in the same file, so usually the overridden pattern will be imported. This functionality is used to modify standard BMC-produced patterns in the field.

It is not an error to name a non-existent pattern in an overrides clause.

Since other patterns might rely on the behaviour of the overridden pattern, the overriding pattern should take care to construct the same form of data model as the overridden pattern.

Requires

requires pattern_name [ , pattern_name ... ];

Declares that this pattern requires the effects of the other named pattern in some way.

The named pattern must be previously declared in the file, either with a pattern declaration, or via an import. It is most useful to declare requirements for patterns in other files, so usually the required pattern will be imported.

Constants

Following the overview is an optional constants section. Each constant takes the form:

\[ name := expression; \]

The name is a valid variable name; the expression is a literal or any expression that depends only on other constants previously declared in the pattern. Named constants are available in the subsequent trigger, body and removal sections. Constant values are not available outside the pattern in which they are declared.

Constants should be used for values that are likely to be changed by pattern authors, such as regular expressions to extract versions from paths, and so on. They should not be used for values that are expected to be changed by normal users of the pattern — in those situations, a Pattern Configuration block should be used.

For example, this defines two regular expressions to be used in the pattern body:

```plaintext
constants cmd_expr := regex "Xvnc";
    display_expr := regex "\b:\(\([0-9]\)+\)\b";end constants;
```
Referencing pattern constants

From TPL version 1.5 (BMC Atrium Discovery 8.2) it is possible to reference constants defined in one pattern from other patterns.

In the first pattern Tomcat 1.0 the constant Tomcat.si_type is defined.

tpl 1.5 module ApacheFoundation.Tomcat;
pattern Tomcat 1.0''''Define contains constants which we want to use elsewhere.'''  
[...]
    constants
    si_type := 'Apache Tomcat Application Server';
end constants;
[...]
end pattern

In the second code sample, the Tomcat 1.0 pattern is imported and in the ExtendedTomcat 1.0 pattern the Tomcat.si_type is referenced.

tpl 1.5 module ApacheFoundation.ExtendedTomcat;
from ApacheFoundation.Tomcat import Tomcat 1.0;
pattern ExtendedTomcat 1.0''''Use specified constant.'''  
[...]
end pattern

Triggers

Triggers are where the active part of patterns begin. Triggers define the conditions in which the body of the pattern are evaluated. From the point of view of pattern authors, conditions are based on data events - creation or modification of nodes in the datastore. Triggers take the form:

```
triggers := [ ... ]
[ on name := [ relationship ] node_kind [ change_kind ]
  [ where trigger_condition ] ] end triggers;
```

In a simple example:

```
triggers on process := DiscoveredProcess where cmd matches regex "Xvnc";
```

The name assigned to the trigger is made available as a variable in the pattern body.

Trigger conditions

Trigger conditions consist of a subset of the more general where clauses available in searches (described later in the section on search expressions). The only operations supported are simple attribute comparisons of the form:

```
attribute_name operator [ expression ]
```

and combinations of such comparisons using and and or.

The only operators available are:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact equality</td>
<td>Matches exactly.</td>
</tr>
</tbody>
</table>


To avoid "greedy" triggers that trigger on almost all data, not expressions are not permitted in triggers.

Expressions must be literal values, constants, or expressions based entirely on literal values and constants.

**Trigger kind**

By default, patterns only trigger when a node is created. In some circumstances, it might be necessary to only trigger when a node is modified, not when it is created. It might also be necessary to trigger if a node is superseded / destroyed.

Patterns can trigger when a node is confirmed to exist, even if it is not modified. Such a pattern triggers when a pattern confirms the existence of a node using one of the Model node assertion functions and, for Host nodes, when a Host is confirmed to exist during a scan.

Triggers can specify a comma-separated set of "change kinds". The kinds are created, confirmed, modified, and destroyed.

For example:

```plaintext
triggers on process := SoftwareInstance confirmed where type matches regex
```

The pattern will only trigger when a matching Software Instance is confirmed to exist, not when a matching Software Instance is first created.

More than one change kind can be selected by separating them with commas. For example:

```plaintext
triggers on process := DiscoveredProcess created, modified where cmd matches
```

The default change kind is created.

When triggering on a node being modified, great care must be taken if the triggering node is modified in the pattern, since the modification inside the pattern causes the pattern to trigger again.

**Relationship triggers**

To trigger on relationships rather than nodes, simply use a prefix of relationship to the kind in the trigger:

```plaintext
triggers // Trigger whenever a HostContainment relationship is created
```

A trigger on a relationship can of course include a where clause.
Existing data
Patterns normally only trigger when data in the datastore changes (or is confirmed by another pattern for the confirmed trigger) — if nodes matching a pattern's trigger conditions already exist at the time the pattern is activated, the pattern will not execute for those nodes. Patterns can be executed against existing nodes using Executing patterns manually.

Body
Once a pattern has triggered, its body is executed. The body consists of assignments, conditionals, loops, function calls, and termination statements.

The statements in the pattern body are not guaranteed to be executed in strict sequence. The system guarantees that dependencies between statements are properly resolved, but statements with no dependencies can be executed out of order or concurrently.

- 1 Assignments
- 2 Conditionals
- 3 Loops
- 4 Body termination
- 5 Value expressions
- 6 Logical expressions
- 7 Node scoped values
- 8 Lists
- 9 Functions
- 10 String interpolation
- 11 Slicing
- 12 Search expressions

Assignments
Assignments take the form:

name := expression;

Where the name must be a valid variable name and the expression must adhere to one of the valid expression kinds described in this section.

It is an error to assign to a name that has been declared as a constant in the same pattern.

Variables are dynamically typed according to the type of the expression. It is not an error to rebind a variable to a value with a different type to its previous value.

void

The void keyword is added in TPL 1.6/BMC Atrium Discovery 8.3. Assigning this to an attribute in the model deletes the attributes. The statement:
node['attribute_name'] := void;

is the equivalent of:

del node ['attribute_name']

**Conditionals**

There is just one kind of conditional statement: the *if*:

```plaintext
if expression then
...
[elif expression then
...
]
[else
...
]
end if;
```

There can be any number of *elif* clauses; the *else* clause is optional.

The expression can have any type — the values *false*, *none*, empty string, empty list, empty set, and numeric zero are considered false; all other values are considered true.

**Loops**

There is only one looping construct, the *for* loop:

```plaintext
for variable in expression do
...
end for;
```

The expression must be a list or set. It is a runtime error if it is not. The loop body executes for each item in the list or set; in each iteration the variable name is set to the corresponding list / set member.

Looping can be terminated early with a *break* statement, or the next iteration started before the end of the loop body with the *continue* statement. For example:

```plaintext
for item in collection do
  if item = 5 then
    break;
  elif item = 10 then
```
continue;
end if;
doSomething(item);
end for;

On exit from a loop, the loop variable is no longer valid. The loop variable cannot be modified in the body of the loop.

**Body termination**

After triggering, it is common for a body to use some "existence criteria" to confirm that the entity modeled by the pattern does indeed exist. It can be useful to terminate the execution of the pattern body prematurely, using the `stop` statement, for example:

```
if not existence_condition then
    stop;
end if;
```

If at all possible, patterns should use trigger conditions to avoid triggering when not necessary, rather than using `stop` in a condition. It is much more efficient to avoid triggering a pattern than to trigger and then stop. If this cannot be avoided then the stop condition should be tested as early as possible especially if additional discovery can be avoided by doing so otherwise considerable load might be put on the system.

**Value expressions**

There are a number of different kinds of expressions used in assignments and other declarations. Simple value expressions consist of literal values as described in the section on Common declarations and variable names.

Numeric values can be involved in arithmetic expressions. Unary expressions prefix a value:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>Negates the value</td>
</tr>
<tr>
<td>+</td>
<td>Has no effect to numeric arguments.</td>
</tr>
<tr>
<td>~</td>
<td>Bitwise inverse</td>
</tr>
</tbody>
</table>

Infix arithmetic operations are the following:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Multiplication</td>
</tr>
<tr>
<td>/</td>
<td>Division</td>
</tr>
<tr>
<td>%</td>
<td>Modulo</td>
</tr>
<tr>
<td>+</td>
<td>Addition</td>
</tr>
<tr>
<td>-</td>
<td>Subtraction</td>
</tr>
<tr>
<td>&lt;&lt;</td>
<td>Bitwise left shift</td>
</tr>
</tbody>
</table>
Bitwise right shift
Bitwise and
Bitwise exclusive or
Bitwise or

The * / % operators take binding precedence over + and -, which take precedence over << and >>, which take precedence over the other operators. Parentheses can be used to group sets of operations to override the precedence rules.

String values can be concatenated using the + infix operator.

**Logical expressions**

Logical expressions are expressions that evaluate to true or false. In boolean operations, values are considered true or false according to the same rules as for if statements, as described in section **Conditionals**. Logical expressions can be combined with the following operators:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a &lt; b</td>
<td>For numeric values, numeric comparisons. For strings, character (lexicographical) comparisons.</td>
</tr>
<tr>
<td>a &lt;= b</td>
<td>Containment tests. True if a is / is not in list or set b.</td>
</tr>
<tr>
<td>a &gt; b</td>
<td>Sub-word and sub-string tests. a and b must be strings, except that a can be none, in which case the test result is false.</td>
</tr>
<tr>
<td>a &gt;= b</td>
<td>True if string a matches regular expression b. If a is none, the test result is false.</td>
</tr>
<tr>
<td>a = b</td>
<td>True if both a and b are; false otherwise.</td>
</tr>
<tr>
<td>a &lt;&gt; b</td>
<td>True if a is considered false; false otherwise.</td>
</tr>
<tr>
<td>a in b</td>
<td>True if a or b or both are true; false if they are both false.</td>
</tr>
</tbody>
</table>

Binding precedence for the operators follows the order they are given above. The operators before not are grouped together with the highest precedence, followed by not, then and, then or.

**Node scoped values**

Variables containing datastore nodes can be used to access the nodes' attributes, using the scoping dot character. For example:

```plaintext
process := // get a DiscoveredProcess node from somewhere
cmd_and_args := process.cmd + " " + process.args;
```

An attempt to access a non-existent attribute results in none.
Key expressions can also be used to access values from related nodes:

\[
\text{method} := \text{process.}\#\text{Member:List:List:ProcessList.discovery\_method;}
\]

Since key expression traversals can find more than one target node, key expressions always return list values. As a consequence, accessing a non-existent traversal key expression results in an empty list.

Attributes of nodes can also be set using the same kind of dot scoping. For example:

\[
\text{host} := \text{model.host(process);} \\
\text{if set\_virtual then} \\
\text{host.virtual} := \text{true;} \\
\text{end if;}
\]

Any attribute name can be set in this way, regardless of whether or not the name is defined in the taxonomy. The attribute is stored with the type of the assigned value, even if the taxonomy specifies that it should have a different type.

In BMC Atrium Discovery the naming convention for attributes is to use only the lowercase letters a to z with underscores (_) to separate words. Attribute names must not start or end with a space. Attribute names in searches can only be specified using the same characters. If you use other characters, for example a hyphen (-), your search will cause an error.

⚠️ **Attribute names cannot be changed**

Once an attribute has been created, you cannot change its name.

**Lists**

Literal lists are declared using [ and ] characters, with a comma-separated sequence of expression inside. e.g.

\[
\text{values} := [\text{"one"}, \text{"two"}, \text{"three"}];
\]

Lists can be dereferenced using the [] operator:

\[
\text{first} := \text{values}[0];
\]

List indexing starts from zero.

Lists can be concatenated together with the + operator. The result of concatenating two lists is a new list; the original lists are left unchanged.

From TPL 1.2, items can be appended to lists with the list.append function:
values := [ "one", "two" ];
list.append(values, "three")

You can also create a list using the list.list function:

values := list.list ( "one", "two", "three" );

Functions

Functions are used to interact with discovery, the data store, and external systems. Functions can be used as statements or as expressions, depending on whether there is a return value. Functions that return a value can be used as statements, in which case the return value is discarded.

Functions are defined outside of the pattern language. Function names are usually scoped, using the dot character to separate the scope from the function name. The scope is used to group related functions together. The currently defined scopes are: log, text, number, regex, xpath, table, discovery, model, email, time, and inference, plus the global scope.

Functions are called as follows

scope.name(arguments)

Function arguments are comma-separated sequences of expressions or assignments. Assignment-style arguments are used to provide named arguments. There are three different ways that functions can process their arguments.

- **Positional arguments**
  In functions with positional arguments, the order in which arguments are provided is significant. Arguments can be optional. Optional arguments are populated in the order they are specified — if a function takes two optional arguments, it is impossible to set the value of the second without also providing a value for the first.

- **Positional and named arguments**
  Functions with positional and named arguments take a number of positional arguments, followed by a number of optional named arguments. Since the named arguments are identified by their names, any of the optional arguments can be set. Arguments are given names using assignment (name := value) syntax.

- **Implicitly and explicitly named arguments**
  Functions whose arguments are all optional named arguments can use implicitly named arguments. In that case, arguments can be provided either with assignment syntax, or can be provided as plain variable names. When plain variable names are used, the argument name is taken from the variable name, and its value from the variable's value. In a function with implicitly named arguments, it is an error to provide non-assignment arguments that are expressions, rather than plain variable names.

The functions are grouped as follows:
- Global scope functions
- Log functions
- Text functions
- Number functions
- Regex functions
- XPath functions
- Table functions
- Discovery functions
- Model functions
- Email functions
- Time functions
- Inference functions

All functions are listed on the Functions page along with a one line overview of each.

**String interpolation**

Unqualified literal strings take part in string interpolation. Values to be interpolated are surrounded by `%` characters. Values are accessed from the pattern’s variables and constants. For example, in:

```plaintext
foo := "world";
bar := "Hello %foo%!";
```

The variable `bar` is set to the string "Hello_world!".

Node scoped variables can also be interpolated. The above command and args example can be implemented with an interpolation:

```plaintext
cmd_and_args := "%process.cmd% %process.args%";
```

To embed a literal percentage symbol in a string, use `%%`

```plaintext
increase := 75;
report := "The size has increased by %increase% %%";
```

Values interpolated into strings must be strings or numbers.

Sometimes it is useful to perform interpolation on a pre-existing string, particularly to specify patterns in constants. This can be achieved with an `expand` expression, e.g.

```plaintext
template := raw "db_%instance%_1";
db_name := expand(template, instance := some_node.instance_name);
```

The use of the `raw` string suppresses the interpolation that would normally occur in setting the `template` variable. The values to interpolate into the string in the `expand` are implicitly and explicitly named arguments as described in the section on `functions` above.
Slicing

Slicing can be applied to lists and strings and it works in the same way as Python. So there are a couple of forms:

```python
value[lowerbound:upperbound]
```

which will start from index `lowerbound` (inclusive) to `upperbound` (exclusive). If the bound is not specified it will represent the start (lower) or end (upper) of the value. It is also possible to use negative values which is equivalent to writing `size(value) + bound`, that is, `value[:size(value)]` is equivalent to `value[:size(value) - 1]`. For example:

```python
value := 'an example string'
```

```python
value[3:] is 'example string'
value[:3] is 'an '
value[-3:] is 'ing'
value[:3] is 'an example str'
```

The second form takes a step value (which if not present is equivalent to a step of 1).

```python
value[::2] is 'a xml tig'
value[::3] is 'aemetn'
value[::-1] is 'gnirts elpmaxe na'
```

Search expressions

Searches using the BMC Atrium Discovery search service are valid expressions in pattern bodies, with a number of minor modifications as described in Search expressions.

Functions

This page provides a list of all TPL functions and is grouped by function type.

Global scope functions

- `size` — Returns the number of items in the collection, where the collection is a list, or a set of nodes retrieved with a search expression.

Log functions

- `log.debug` — Log the given message with a debug level message. The log messages that are output automatically include the name of the pattern performing the log action.
- `log.info` — Log the given message with an info level message. The log messages that are output automatically include the name of the pattern performing the log action.
• **log.warn** — Log the given message with a warn level message. The log messages that are output automatically include the name of the pattern performing the log action.
• **log.error** — Log the given message with an error level message. The log messages that are output automatically include the name of the pattern performing the log action.
• **log.critical** — Log the given message with a critical level message. The log messages that are output automatically include the name of the pattern performing the log action.

**Text functions**

• **text.lower** — Returns the lower-cased version of the string argument.
• **text.upper** — Returns the upper-cased version of the string argument.
• **text.toNumber** — Converts its string argument into a number.
• **text.replace** — Returns a modified version of the string formed by replacing all occurrences of the string old with new.
• **text.join** — Returns a string containing all items in a list of strings joined with the specified separator.
• **text.split** — Returns a list consisting of portions of the string split according to the separator string, where specified.
• **text.strip** — Strips unwanted characters from the start and end of the given string.
• **text.leftStrip** — Equivalent to text.strip, but only strips from the left side of the string.
• **text.rightStrip** — Equivalent to text.strip, but only strips from the right side of the string.
• **text.hash** — Returns a hashed form of the string, generated with the MD5 hash algorithm.
• **text.ordinal** — Returns the ordinal value of the string argument.

**Number functions**

• **number.toChar** — Converts the integer number in the ASCII range to a character.
• **number.toText** — Converts the integer number to a text form.
• **number.range** — Generate a list containing 0 to number - 1.

**Regex functions**

• **regex.extract** — Returns the result of extracting the regular expression from the string, optionally with a substitution expression and a specified result if no match is found.
• **regex.extractAll** — Returns a list containing all the non-overlapping matches of the pattern in the string.

**XPath functions**

• **xpath.evaluate** — Returns the result of evaluating the xpath expression against the XML string.
• **xpath.openDocument** — Returns the DOM object resulting from parsing the XML string.
• **xpath.closeDocument** — Closes the DOM object resulting from xpath.openDocument.
Table functions

- **table** — Creates a new table.
- **table.remove** — Removes the specified key from the specified table.

Discovery functions

- **Discovery action functions**
  - **discovery.fileGet** — Retrieves the specified file.
  - **discovery.fileInfo** — Retrieves information about the specified file, but not the file content.
  - **discovery.getNames** — Performs a DNS lookup on the IP address and returns a list of FQDN strings. Finds the DiscoveryAccess from the target node and attaches the list of FQDNs to it.
  - **discovery.listDirectory** — Retrieves the directory listing of the directory specified by the path on the specified target.
  - **discovery.listRegistry** — Returns a list of the registry entries of the registry key specified by the key_path.
  - **discovery.registryKey** — Retrieves a registry key from a Windows computer.
  - **discovery.wmiQuery** — Performs a WMI query on a Windows computer.
  - **discovery.wbemQuery** — Performs a WBEM query on the target and returns a list of DiscoveredWBEM DDD nodes.
  - **discovery.wbemEnumInstances** — Performs a WBEM query on the target and returns a list of DiscoveredWBEMInstance DDD nodes.
  - **discovery.runCommand** — Returns a DiscoveredCommandResult node containing the result of running the specified command.
  - **discovery.snmpGet** — Performs an SNMP query on the target and returns a DiscoveredSNMP node.
  - **discovery.snmpGetTable** — Performs an SNMP query that returns a table on the target.

- **Discovery data manipulation functions**
  - **discovery.process** — Returns the process node corresponding to the source node, which must be a ListeningPort or NetworkConnection node.
  - **discovery.children** — Returns a list of the child processes for the given DiscoveredProcess node.
  - **discovery.descendants** — Returns a list consisting of the children of the given DiscoveredProcess node, and recursively all of the children's children.
  - **discovery.parent** — Returns the parent process for the given DiscoveredProcess node.
  - **discovery.allProcesses** — Returns a list of all processes corresponding to the directly discovered data source node.
• **discovery.access** — Returns the Discovery Access node for the source node, which must be a directly discovered data node.

**Binary functions**

• **binary.toHexString** — Returns the given binary value as a hex string, that is, two hex digits per byte.
• **binary.toIPv4** — Returns the given binary value as the text representation of an IPv4 address.
• **binary.toIPv4z** — Returns the given binary value as the text representation of an IPv4 address with a zone index.
• **binary.toIPv6** — Returns the given binary value as the text representation of a canonical IPv6 address.
• **binary.toIPv6z** — Returns the given binary value as the text representation of a canonical IPv6 address with zone index.
• **binary.toMACAddress** — Returns the given binary value as the text representation of a MAC address.
• **binary.toValue** — Converts the given binary value into the specified format.
• **binary.toWWN** — Returns the given binary value as the text representation of a WWN value.

**Model functions**

• **model.addContainment** — Adds the containees to the container by creating suitable relationships between the nodes.
• **model.setContainment** — Equivalent to addContainment, except unconfirmed relationships are removed at the end of the pattern body
• **model.destroy** — Destroy the specified node or relationship in the model.
• **model.withdraw** — Removes the named attribute from the node.
• **model.setRemovalGroup** — Add the specified node or nodes to a named removal group.
• **model.anchorRemovalGroup** — Specify an anchor node for a named removal group.
• **model.suppressRemovalGroup** — Suppress removal of the named removal group.
• **model.host** — Returns the Host node corresponding to the given node.
• **model.hosts** — Returns a list of all the Host nodes corresponding to the given node.
• **model.findPackages** — Traverses from the node, which must be a Host or a directly discovered data node, and returns a set of all Package nodes that have names matching the provided list of regular expressions.
• **model.addDisplayAttribute** — Adds a named attribute, or a list of named attributes to the additional attributes displayed in a node view.
• **model.removeDisplayAttribute** — Removes a named attribute, or a list of named attributes from the additional attributes displayed in a node view.
Related functions

- related.detailContainer — Returns the Software Component, Software Instance, or Business Application Instance node containing to the given node.
- related.host — Returns the Host node corresponding to the given node.

Email functions

- mail.send — Sends an email.

Time functions

- time.current — Returns the current time.
- time.delta — Creates a time delta that can be added to or subtracted from a time.
- time.parseLocal — The time is converted according to the appliance's time zone and daylight saving setting.
- time.parseUTC — The time is assumed to already be UTC, and is not adjusted for timezones or daylight saving time.
- time.formatLocal — Formats a time into a human-readable string.
- time.formatUTC — Formats a time into a human-readable string.
- time.toTicks — Converts a time or time delta into the DCE ticks format.
- time.fromTicks — Converts DCE ticks into a time.
- time.deltaFromTicks — Converts DCE ticks into a time delta.

Inference functions

- inference.associate — Create associate inference relationship(s) from the specified node(s) to the inferred node.
- inference.contributor — Create contributor inference relationship(s) from the specified node(s) to the inferred node, for attribute names specified in the contributes list.
- inference.primary — Create primary inference relationship(s) from the specified node(s) to the inferred node.
- inference.relation — Create relation inference relationship(s) from the specified node(s) to the inferred relationship.
- inference.withdrawal — Create withdrawal inference relationship(s) from the specified node(s) to the inferred node, indicating the withdrawal of the withdrawn attribute name.
- inference.destruction — When destroying a node, indicate that the source node was responsible for its destruction.

System functions

- system.getOption — Takes the name of a BMC Atrium Discovery system option and returns the value.
Global scope functions
The following functions are in the global scope. They are called by simply giving the function name. They take positional arguments.

- `size`

`size( collection )`
Returns the number of items in the collection, where the collection is a list, or a set of nodes retrieved with a search expression.

Log functions
The following functions are for logging. Each takes a single positional argument:

- `log.debug`
- `log.info`
- `log.warn`
- `log.error`
- `log.critical`

`log.debug( string )`
Log the given message with a debug level message. The log messages that are output automatically include the name of the pattern performing the log action.

`log.info( string )`
Log the given message with an info level message. The log messages that are output automatically include the name of the pattern performing the log action.

`log.warn( string )`
Log the given message with a warn level message. The log messages that are output automatically include the name of the pattern performing the log action.

`log.error( string )`
Log the given message with an error level message. The log messages that are output automatically include the name of the pattern performing the log action.

`log.critical( string )`
Log the given message with a critical level message. The log messages that are output automatically include the name of the pattern performing the log action.

Text functions
The following are `text` functions that operate on text strings. They take positional arguments.

- `text.lower`
- `text.upper`
- `text.toNumber`
- `text.replace`
- **text.join**
- **text.split**
- **text.strip**
- **text.leftStrip**
- **text.rightStrip**
- **text.hash**
- **text.ordinal**

**text.lower( string )**

Returns the lower-cased version of the `string` argument.

**text.upper( string )**

Returns the upper-cased version of the `string` argument.

**text.toNumber( string [, base ])**

Converts its string argument into a number. By default, the number is treated as a base 10 value; if the optional `base` is provided, the number is treated as being in the specified base. Bases from 2 to 36 are supported. Bases larger than 10 use the letters a through z to represent digits.

**text.replace( string , old , new )**

Returns a modified version of the string formed by replacing all occurrences of the `string old` with `new`. For example, `text.replace("hello world", "o", "OO")` returns "hellOO wOOrld".

**text.join( list , " separator ")**

Returns a string containing all items in a list of strings joined with the specified separator. Returns an empty string when the list is empty. Where the list only contains one item, it returns the unmodified item; no separator is appended.

For example:

```python
input_list

One
Two
Three
Four
```

**text.join(input_list, ", ") returns:**

```python
One, Two, Three, Four
```

**text.split( string , [ separator ])**
Returns a list consisting of portions of the string split according to the separator string, where specified. The separator does not appear in the output string. For example, `text.split("Split this string", " ")` results in the list `"Split", "this", "string"`.

If the separator is not specified, the split will occur on all whitespace characters.

If the text is not split because the separator does not occur in the string, the function returns a list containing the whole string as its only member.

The separator is optional in BMC Atrium Discovery 8.3 and later.

- `text.strip(string [, characters ] )` — (7.1 and later)
  Strips unwanted characters from the start and end of the given string. If the optional `characters` is not provided, white space characters are stripped; if `characters` is provided, it contains the characters that are to be removed.

- `text.leftStrip(string [, characters ] )` — (7.1 and later)
  Equivalent to `text.strip`, but only strips from the left side of the string.

- `text.rightStrip(string [, characters ] )` — (7.1 and later)
  Equivalent to `text.strip`, but only strips from the right side of the string.

- `text.hash(string)`
  Returns a hashed form of the string, generated with the MD5 hash algorithm.

- `text.ordinal(string)`
  Returns the ordinal value of the `string` argument. The string must be one character in length.

  For example the ordinal value of `A` is 65.

**Number functions**

The following are number functions that operate on numeric arguments. They take positional or keyword arguments.

- `number.toChar`
- `number.toText`
- `number.range`

- `number.toChar(number)`
  Converts the integer number in the ASCII range to a character. If the value is outside the ASCII range, it returns `none`.

- `number.toText(number [, base [, width ]])`
  Converts the integer number to a text form. The default is to use the base 10 representation. If the optional base argument is given, conversion uses that base instead. Only bases 8, 10 and 16 are supported. If the optional width argument it given, the output is padded with leading zeros to have at least the specified width. If the number is too large, the output can exceed the specified width.

- `number.range(number)`
  Generate a list containing 0 to number - 1. If number is less than 1 the list will be empty. For example, `number.range(5)` results in the list `[0, 1, 2, 3, 4]`. 
### Regex functions

The following are regex functions. They take positional arguments.

- `regex.extract`
- `regex.extractAll`

**regex.extract( string, expression [, substitution ] [, no_match])**

Returns the result of extracting the regular expression from the string, optionally with a substitution expression and a specified result if no match is found. Returns an empty string, or the string specified in `no_match` if the expression does not match. If the string specified was none, an empty string is returned.

`no_match` was introduced in BMC Atrium Discovery 8.3.

The optional substitution expression is used to store groups that match the search expression. The first group is stored in \1, the second in \2, and so forth. For example:

```java
// Extract the version using group 1 of the regular expression.
version := regex.extract(process.cmd, path_regex, raw "\1", no_match := 'No match.');
```

A versioning string is of the form "Version 2.7.182 patch 69". To extract the patch number, the following function is used:

```java
patch := regex.extract(cmdResult.result, regex 'patch (\d+)',
                      raw '\1', no_match := 'Unsupported.');
```

The expression `patch (\d+)` matches patch 69 in the versioning string and it contains the first group matching \1 which is 69. If the versioning string was "Version 2.7.182 patch 69 patch 70" then the second group matching \2 is 70.

If the versioning string was "Version 2.7.182", the function would return "Unsupported."

**regex.extractAll( string, pattern )**

Returns a list containing all the non-overlapping matches of the pattern in the string. If the pattern contains more than one group, returns a list of lists containing the groups in order.

### XPath functions

The following are xpath functions. They take positional arguments.

- `xpath.evaluate`
- `xpath.openDocument`
- `xpath.closeDocument`
xpath.evaluate( string , expression )
Returns the result of evaluating the xpath expression against the XML string. Returns a list of strings containing the selected values. The result is always a list, even if the expression is guaranteed to always return just one result.

The expression must be one that returns tag attributes or body text - expressions returning DOM nodes will result in an error that terminates execution of the pattern.

xpath.openDocument( string )
Returns the DOM object resulting from parsing the XML string. The DOM object returned is suitable for passing to xpath.evaluate.

Parsing the XML once is considerably more efficient if you wish to call xpath.evaluate on the same XML.

You should use xpath.closeDocument once xpath evaluation is complete or the parsed DOM will be held in memory for some time.

xpath.closeDocument( DOM object )
Closes the DOM object resulting from xpath.openDocument.

You should close an XML document opened with xpath.openDocument once xpath evaluation is complete or the parsed DOM will be held in memory for some time.

**Table functions**

Patterns can create dynamic *tables* that provide a mapping from keys to values. Dynamic tables were introduced in TPL 1.2. The static form is supported by all versions of TPL, see Static Tables for more information.

- *table*
- *table.remove*

**table( [ parameters ] )**

Creates a new table. With no parameters, creates an empty table. If parameters are given, initializes the table with items where the keys are the parameter names and the values are the parameter values.

Table items are accessed with square brackets, and a *for* loop iterates over the table keys. For example:

```
things := table();
things["one"] := 1;
things["two"] := 2;
for key in things do  value := things[key]; log.info("%key% -> %value%"); end
```

**table.remove( table, key )**

Removes the specified key from the specified table.
Discovery functions

Discovery action functions
The following discovery functions retrieve information from discovery targets. They take positional arguments. They return `None` if an error occurs in performing the discovery.

- discovery.fileGet
- discovery.fileInfo
- discovery.getNames
- discovery.listDirectory
- discovery.listRegistry
- discovery.registryKey
- discovery.wmiQuery
- discovery.wbemQuery
- discovery.wbemEnumInstances
- discovery.runCommand
- discovery.snmpGet
- discovery.snmpGetTable

Discovery data manipulation functions
The following functions in the discovery module navigate across the stored directly discovered data. They do not perform any discovery actions. They take positional arguments.

- discovery.process
- discovery.children
- discovery.descendents
- discovery.parent
- discovery.allProcesses
- discovery.access

Discovery action functions
The following discovery functions retrieve information from discovery targets. They take positional arguments. They return `None` if an error occurs in performing the discovery.

- discovery.fileGet
- discovery.fileInfo
- discovery.getNames
- discovery.listDirectory
- discovery.listRegistry
- discovery.registryKey
- discovery.wmiQuery
- discovery.wbemQuery
- discovery.wbemEnumInstances
- discovery.runCommand
- discovery.snmpGet
discovery.snmpGetTable
discovery.fileGet(target, filename [, win64_redirect = true|false])
Retrieves the specified file. target is a node used to identify the discovery target, either a directly discovered data node, or a Host node. Requires PRIV_CAT to be defined to retrieve files not readable by the current user.

64 bit Windows hosts have the following system directories for holding executables, DLLs, and so forth:

- %windir%\system32 — 64 bit executables and DLLs
- %windir%\SysWOW64 — 32 bit executables and DLLs

win64_redirect is an optional flag which enables you to specify whether the search for the file should take place in the 32 bit directory if not found in the 64 bit directory. TRUE means search the 32 bit directory, FALSE means do not search the 32 bit directory if the file is not found in the 64 bit directory. The default is TRUE.

Filenames that do not start with %windir%\system32 or %systemroot%\system32 are never checked for redirection.

On non-Windows and 32 bit Windows systems win64_redirect has no effect.

The win64_redirect flag is new in 8.2.

Returns a DiscoveredFile node.

The target node can be one of the following:

- Host
- DiscoveryAccess
- DiscoveredDirectoryEntry
- DiscoveredFQDN
- DiscoveredHBA
- DiscoveredListeningPort
- DiscoveredNetworkConnection
- DiscoveredNetworkInterface
- DiscoveredProcess
- DiscoveredSNMPRow
- DiscoveredWMI
- SQLResultRow
- DirectoryListing
- DiscoveredPackages
- DiscoveredSNMPTable
- DiscoveredWMIQuery
discovery.fileInfo( target , filename [, win64_redirect =true|false]) — (7.2 and later)

Retrieves information about the specified file, but not the file content. This is useful if the file is a binary file or particularly large.

64 bit Windows hosts have the following system directories for holding executables, DLLs, and so forth:

- %windir%\system32 — 64 bit executables and DLLs
- %windir%\SysWOW64 — 32 bit executables and DLLs

win64_redirect is an optional flag which enables you to specify whether the search for the file should take place in the 32 bit directory if not found in the 64 bit directory. TRUE means search the 32 bit directory, FALSE means do not search the 32 bit directory if the file is not found in the 64 bit directory. The default is TRUE.

Filenames that do not start with %windir%\system32 or %systemroot%\system32 are never checked for redirection.

On non-Windows and 32 bit Windows systems win64_redirect has no effect.

The win64_redirect flag is new in 8.2.

Returns a DiscoveredFile node with no content attribute.

Retrieves the specified file.

discovery.getNames( target, ip_address )

Performs a DNS lookup on the IP address and returns a list of FQDN strings. Finds the DiscoveryAccess from the target node and attaches the list of FQDNs to it. The target can be a DDD node or a Host. If no names are found it returns an empty list. If an error occurs the discovery.getNames function returns None.

discovery.listDirectory( target , path [, win64_redirect =true|false])
Retrieves the directory listing of the directory specified by the path on the specified target. You cannot use wildcards in the path. Returns a list of Discovered Directory Entry nodes.

64 bit Windows hosts have the following system directories for holding executables, DLLs, and so forth:

- `%windir\system32` — 64 bit executables and DLLs
- `%windir\SysWOW64` — 32 bit executables and DLLs

win64_redirect is an optional flag which enables you to specify whether the search should take place in the 32 bit directory if not found in the 64 bit directory. TRUE means search the 32 bit directory, FALSE means do not search the 32 bit directory if the directory is not found beneath the 64 bit directory. The default is TRUE.

Paths that do not start with `%windir\system32` or `%systemroot\system32` are never checked for redirection.

On non-Windows and 32 bit Windows systems win64_redirect has no effect.

The win64_redirect flag is new in 8.2.

If the call fails, it returns none and records the failure reason in the Directory Listing node.

Retrieves the specified file.

Restrictions

1. On Windows systems, owner and group information is not recovered.
2. Group names which include spaces are not supported.
3. Config.sys is not listed as a system file.
4. Directory Listing nodes are not created for empty Windows directories.

discovery.listRegistry( target , key_path [, win64_redirect =true|false]) — (8.1 and later )

Returns a list of the registry entries of the registry key specified by the key_path.

64 bit Windows hosts have the following registry keys:

- HKEY_LOCAL_MACHINE\SOFTWARE — 64 bit values
- HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node — 32 bit values

win64_redirect is an optional flag which enables you to specify whether the search for the registry entries should take place in the 32 bit section if not found in the 64 bit section. TRUE means search the 32 bit section, FALSE means do not search the 32 bit section if the registry entries are not found in the 64 bit section. The default is TRUE.

On non-Windows and 32 bit Windows systems win64_redirect has no effect.

The win64_redirect flag is new in 8.2.
Returns a list of DiscoveredRegistryEntry nodes. If the call fails, it returns none and records the failure reason in the RegistryListing node.

discovery.registryKey( target , key [, win64_redirect =true|false])

Retrieves a registry key from a Windows computer.

64 bit Windows hosts have the following registry keys:

- HKEY_LOCAL_MACHINE\SOFTWARE — 64 bit values
- HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node — 32 bit values

win64_redirect is an optional flag which enables you to specify whether the search for the registry key should take place in the 32 bit section if not found in the 64 bit section. TRUE means search the 32 bit section, FALSE means do not search the 32 bit section even if the key is not found in the 64 bit section. The default is TRUE.

On non-Windows and 32 bit Windows systems win64_redirect has no effect.

The win64_redirect flag is new in 8.2.

Returns a DiscoveredRegistryValue node.
Retrieves the specified file.

discovery.wmiQuery( target , query , namespace )

Performs a WMI query on a Windows computer. Returns a list of DiscoveredWMI nodes.
Retrieves the specified file.

discovery.wbemQuery(target, class_name, [ properties ], namespace)

Performs a WBEM query on the target and returns a list of DiscoveredWBEM DDD nodes.

💡 This is not a generic query for any device managed by WBEM; rather, it has been implemented solely as a part of storage discovery. Using it for any other purpose is untested and may provide unexpected results and very likely return extremely large lists of DDD nodes.

Where:

- target — the endpoint being scanned, usually a node kind which can be used to get to the DiscoveryAccess, for example, a host node.
- class_name — the name of the WBEM (CIM) class being queried, for example, CIM_LogicalElement.
- properties — a list of properties to be retrieved from the class. Where a specified property does not exist on the target, it is ignored. The properties list is not case sensitive, for example, if you request stuff and the property is called STUFF or Stuff, it is returned, but in the case in which it is returned is the same as it is on the target.
- The properties list may be empty.
- **namespace** – the WBEM namespace to query, for example, `root/cimv2`. Used with the `class_name` to identify objects of interest.

Returns a list of `DiscoveredWBEM DDD nodes` which hold the results. Each `DiscoveredWBEM node` has a list of attributes corresponding to those specified in the `properties` list, plus a `__PATH` attribute representing the local WBEM instance path for the object. The `__PATH` attribute is returned whether the `properties` list is empty or not.

If the call fails, none are returned.

An additional **WBEM credential** is required which matches the endpoint being scanned. This is not the host credential used to scan the endpoint.

`discovery.wbemEnumInstances(target, class_name, properties, namespace)`

Performs a WBEM query on the target and returns a list of `DiscoveredWBEMInstance DDD nodes`.

⚠️ This is not a generic query for any device managed by WBEM; rather, it has been implemented solely as a part of storage discovery. Using it for any other purpose is untested and may provide unexpected results and very likely return extremely large lists of DDD nodes.

Where:

- **target** – the endpoint being scanned, usually a node kind which can be used to get to the `DiscoveryAccess`, for example, a host node.
- **class_name** – the name of the WBEM (CIM) class being queried, for example, `CIM_LogicalElement`.
- **properties** – a list of properties to be retrieved from the class. Where a specified property does not exist on the target, it is ignored. The `properties` list is not case sensitive, for example, if you request `stuff` and the property is called `STUFF` or `Stuff`, it is returned, but in the case in which it is returned is the same as it is on the target.
- **namespace** – the WBEM namespace to query, for example, `root/cimv2`. Used with the `class_name` to identify objects of interest.

Returns a list of `DiscoveredWBEM DDD nodes` which hold the results. Each `DiscoveredWBEM node` has a list of attributes corresponding to those specified in the `properties` list, plus a `__PATH` attribute representing the local WBEM instance path for the object. The `__PATH` attribute is returned whether the `properties` list is empty or not.

If the call fails, none are returned.
An additional **WBEM credential** is required which matches the endpoint being scanned. This is not the host credential used to scan the endpoint.

discovery.runCommand(target, command)

Returns a DiscoveredCommandResult node containing the result of running the specified command.

Retrieves the specified file.

The following code examples shows `discovery.runcommand` being used to determine the publisher of a Java virtual machine. This comes from the TKU Java VM pattern.

```
publisher_command := '"javacmd_path" -version';

// Run java -version and extract the publisher
ran_pub_cmd := discovery.runCommand(host, publisher_command);
...
...
if ran_pub_cmd then
    log.debug("Publisher command output: \$ran_pub_cmd.result\$"/
    cmd_publisher := regex.extract(ran_pub_cmd.result,
        regex'(?i)(hotspot|gnu|ibm|bea|sun)', raw'\$1');
end if;
```

⚠️ **Possible post-processing required on UNIX/Linux hosts**

When a pattern executes a command on a UNIX/Linux host, the result returned can contain control characters used to display colours and other formatting. The pattern should not rely on being able to use the result without some additional processing such as regular expression matching and extraction.

discovery.snmpGet(target, oid_table, [binary_oid_list])

Performs an SNMP query on the target and returns a DiscoveredSNMP node. The **oid_table** is a **table** that lists the SNMP OIDs to retrieve, mapped to the attribute names to set. Binary data (SNMP octet strings with embedded nulls) such as IP or MAC addresses must be flagged before they can be recovered correctly. The optional **binary_oid_list** parameter (available in BMC Atrium Discovery 10.0 and later) is a list of OIDs that contain binary data. For example:

```
table oids 1.0"1.3.6.1.2.1.1.1" -> "one";"1.3.6.1.2.3.4.5" -> "two";end ts
```

DiscoveredSNMPnode := discovery.snmpGet(target, oids, ["1.3.6.1.2.3.4.5"])
When binary data is successfully retrieved, you can only use the binary functions to process the data.

The call succeeds if any of the requested OIDs are available; the attributes corresponding to any OIDs that were not available are not set on the DiscoveredSNMP node. If none of the OIDs are available, the discovery request fails and the snmpGet function returns none.

The target node can be one of the following:

- DeviceInfo
- DirectoryListing
- DiscoveryAccess
- FileSystemList
- FQDNList
- HBAInfoList
- Host
- HostInfo
- IntegrationResult
- InterfaceList
- NetworkConnectionList
- NetworkDevice
- ProcessList
- RegistryListing
- ServiceList
- SQLResultRow
- DiscoveredAggregatedPorts
- DiscoveredAggregatedPortsList
- DiscoveredATMVirtualCircuit
- DiscoveredATMVirtualCircuitList
- DiscoveredATMVirtualPath
- DiscoveredATMVirtualPathList
- DiscoveredBridge
- DiscoveredBridgeList
- DiscoveredCard
- DiscoveredCardList
- DiscoveredChassis
- DiscoveredChassisList
- DiscoveredCommandResult
- DiscoveredDevicePort
- DiscoveredDevicePortList
- DiscoveredDirectoryEntry
- DiscoveredFile
discovery.snmpGetTable( target , table_oid , column_table , [ binary_oid_list ])

Performs an SNMP query that returns a table on the target. Returns a list of DiscoveredSNMPRow nodes or none if the SNMP query failed. The table_oid parameter specifies the SNMP OID for the table; the column OIDs in the table are mapped to attribute names using the column_table. Not all column OIDs need to be listed in the storage table. If a column OID is not mapped it will be dropped and not stored in the DiscoveredSNMPRow nodes. Similarly, if a mapping for a non-existent column OID is given, it will be ignored. Binary data (SNMP octet strings with embedded nuls) such as IP or MAC addresses must be flagged before they can be recovered correctly. The optional binary_oid_list parameter (available in BMC Atrium Discovery 10.0 and later) is a list of OIDs that contain binary data. For example, to retrieve a table of storage information:

```

table storage_map 1.0"1" -> "index";"2" -> "type";"3" -> "descr";"4" -> "size"

pattern example 1.0
...body   rows := discovery.snmpGetTable(host, "1.3.6.1.2.1.99.99", store
```
OID usage with snmpGetTable

When using snmpGetTable, specifying the OID of a table object, a DiscoveredSNMPRow node is created for every requested value in every row in the table rather than a single node for each row in the table. To avoid this you should use the OID for the entry object. For example, rather than using .1.3.6.1.2.1.2.2 (ifTable) you need to use .1.3.6.1.2.1.2.2.1 (ifEntry).

When binary data is successfully retrieved, you can only use the binary functions to process the data.

Performs an SNMP query on the target and returns a DiscoveredSNMP node.

Discovery data manipulation functions

The following functions in the discovery module navigate across the stored directly discovered data. They do not perform any discovery actions. They take positional arguments.

- discovery.process
- discovery.children
- discovery.descendents
- discovery.parent
- discovery.allProcesses
- discovery.access

**discovery.process( source )**

Returns the process node corresponding to the source node, which must be a ListeningPort or NetworkConnection node.

**discovery.children( process )**

Returns a list of the child processes for the given DiscoveredProcess node. Returns an empty list if there no children or the parameter is not a DiscoveredProcess node.

**discovery.descendents( process )**

Returns a list consisting of the children of the given DiscoveredProcess node, and recursively all of the children’s children.

**discovery.parent( process )**

Returns the parent process for the given DiscoveredProcess node. Returns none if the process has no parent.

**discovery.allProcesses( source )**

Returns a list of all processes corresponding to the directly discovered data source node. Returns an empty list if the source node is not a valid directly discovered data node.
Returns the Discovery Access node for the **source** node, which must be a directly discovered data node. Returns **none** if the source is not valid.

**Binary functions**

The following **binary** functions return a text representation of the given binary data retrieved using `discovery.snmpGet` or `discovery.snmpGetTable`. They return **none** if the value is invalid, for example the wrong size for an IP address, or is not a binary value.

- **binary.toHexString** — Returns the given binary value as a hex string, that is, two hex digits per byte.
- **binary.toIPv4** — Returns the given binary value as the text representation of an IPv4 address.
- **binary.toIPv4z** — Returns the given binary value as the text representation of an IPv4 address with a zone index.
- **binary.toIPv6** — Returns the given binary value as the text representation of a canonical IPv6 address.
- **binary.toIPv6z** — Returns the given binary value as the text representation of a canonical IPv6 address with zone index.
- **binary.toMACAddress** — Returns the given binary value as the text representation of a MAC address.
- **binary.toValue** — Converts the given binary value into the specified format.
- **binary.toWWN** — Returns the given binary value as the text representation of a WWN value.

**binary.toHexString(value)**

Returns the given binary value as a hex string, that is, two hex digits per byte. If the value is invalid, for example, it is not a binary value, it returns **none**.

**binary.toIPv4(value)**

Returns the given binary value as the text representation of an IPv4 address. If the value is invalid, for example, it is the wrong size, it returns **none**.

**binary.toIPv4z(value)**

Returns the given binary value as the text representation of an IPv4 address with a zone index. If the value is invalid, for example, it is the wrong size, it returns **none**.

**binary.toIPv6(value)**

Returns the given binary value as the text representation of a canonical IPv6 address. If the value is invalid, for example, it is the wrong size, it returns **none**.

**binary.toIPv6z(value)**

Returns the given binary value as the text representation of a canonical IPv6 address with zone index. If the value is invalid, for example, it is the wrong size, it returns **none**.

**binary.toMACAddress(value)**

Returns the given binary value as the text representation of a MAC address. If the value is invalid, for example, it is the wrong size, it returns **none**.
**binary.toValue**

`binary.toValue(data, format)`

Converts the given binary value into the specified format. The binary value is typically a value retrieved via SNMP using the binary settings on `discovery.snmpGet` or `discovery.snmpGetTable`.

If the function fails, for example, if the value is invalid, or an unknown format is specified, it returns `none`.

**Format specification**

The format specification is a string of two or more characters. The first character specifies the byte order of the data.

<table>
<thead>
<tr>
<th>Character</th>
<th>Byte order</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>little-endian</td>
</tr>
<tr>
<td>&gt;</td>
<td>big-endian</td>
</tr>
<tr>
<td>!</td>
<td>network (= big-endian)</td>
</tr>
</tbody>
</table>

The remaining characters represent the Python types and sizes.

<table>
<thead>
<tr>
<th>Format</th>
<th>Python type</th>
<th>Standard size</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>no value</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>string of length 1</td>
<td>1</td>
</tr>
<tr>
<td>B or b</td>
<td>integer</td>
<td>1</td>
</tr>
<tr>
<td>?</td>
<td>bool</td>
<td>1</td>
</tr>
<tr>
<td>H or h</td>
<td>integer</td>
<td>2</td>
</tr>
<tr>
<td>I or i</td>
<td>integer</td>
<td>4</td>
</tr>
<tr>
<td>L or l</td>
<td>integer</td>
<td>4</td>
</tr>
<tr>
<td>Q or q</td>
<td>integer</td>
<td>8</td>
</tr>
<tr>
<td>f</td>
<td>float</td>
<td>4</td>
</tr>
<tr>
<td>d</td>
<td>float</td>
<td>8</td>
</tr>
<tr>
<td>s</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>integer</td>
<td></td>
</tr>
</tbody>
</table>

For example:

- "!HI" means convert 6 bytes in network byte order into a 16 bit integer value and a 32 bit integer value.
- "!4H" or "!HHHHH" means convert 8 bytes in network byte order into four 16 bit integer values.
The information in the tables above is taken from the Python documentation.

Example

The FCMGMT-MIB::connUnitPortTable contains a value called connUnitPortFCClassCap which is defined as a 2 octet string (2 bytes). Each bit in this string of bytes indicates the FC class capabilities of the port. To decode this in TPL, you convert the 2 bytes into a number and then use the TPL bit mask operators.

In the following example, the format specifier "!H" means convert 2 bytes in network byte order into a 16 bit integer value:

```python
value := binary.toValue(connUnitPortFCClassCap, "!H");
```

**binary.toWWN(value)**

Returns the given binary value as the text representation of a WWN value. If the value is invalid, for example, it is the wrong size, it returns none.

**Model functions**

**Model node existence functions**

The `model` scope is automatically populated with a function named after each node kind defined in the taxonomy. The functions are used to ensure the existence of corresponding nodes. Each function takes implicitly named arguments of the form:

```python
model.node_kind(attributes...)
```

The function creates or updates a node with the relevant kind, and returns the node. The node's attributes are set to the values given in the function's implicitly named arguments. For example:

```python
key := "My unique key";
name := "My instance number %number%";
special_value := 123;model.SoftwareInstance(key, name, value := special_value + 1);
```

A Software Instance node with the specified key is created or updated. It has attributes key, name and value. If a Software Instance with the given key already exists, it is updated; otherwise a new one is created.

Any attribute name can be set in the named arguments, regardless of whether the names are defined in the taxonomy. The attributes are stored with the types of the assigned values, even if the taxonomy specifies that they should have different types.

Nodes created by patterns should have unique keys, both so they can be identified in subsequent executions of the same pattern, and to facilitate integrations that export the data. Keys should be unique across the entire datastore.
In some cases, it is not possible to generate a key that is unique in any scope. In that case, a
grouped entity is created, containing a count of the number of nodes in the group, with one group
node per Host. If it is possible to separate the groups into a number of sub-groups, the key_group
attribute can be set, in which case one grouping node will be created for each of the different
key_group values.

Node names types and keys contains some advice about choosing appropriate names, types, and
keys for nodes created by patterns.

Each node created or modified by these node functions is automatically given a primary inference
relationship to the node that triggered execution of the pattern, plus additional contributor inference
relationships to represent the provenance of all attributes set on the node. Software Instance and
Business Application Instance nodes are also automatically related to the Host(s) on which they
are running.

Nodes created or modified by these functions are also given a maintainer relationship to the
pattern. When a pattern is removed from the system, all the nodes it has maintainer relationships
to are destroyed.

⚠️ Accidental maintainers

Never use the model.nodekind functions to update attributes on a node that the
pattern is not maintaining. For example, to set an attribute on a Host node, do not do this:

```
my_host := // a Host from somewheremodel.Host(key := my_host.key, color := "red"); // BAD!
```

instead, directly set attributes on the node:

```
my_host := // a Host from somewheremy_host.color := "red"; // Good!
```

Model relationship existence functions

Where additional relationships are required, they can be created explicitly. For each relationship
defined in the taxonomy, a corresponding function is also defined in the model.rel scope. It
takes the form:

```
model.rel.relationship(source_role := source, dest_role := dest, attributes ...)
```

The function creates or updates a relationship between the source and destination nodes. The
parameters are named after the relationship's roles, e.g.

```
si_node := // VMware SoftwareInstancehost_node := // Virtual Hostrel := model.rel.HostContainment(HostContainer := si_node, ContainedHost := host_node);
```
The relationship functions also accept lists of nodes in the roles, in which case, multiple relationships are created/updated between each pair of nodes and the function returns a list of the relationships.

⚠️ **Use of model.rel.Inference should be avoided**

You should avoid using the `model.rel.inference` function. The inference system expects specific attributes on the Inference relationship. If these attributes are missing there might be a traceback whenever the system attempts to update inference information. A warning is logged when you upload a pattern using the `model.rel.inference` function.

Relationships in custom patterns

During the development and testing of a custom pattern, if the pattern creates incorrect relationships between nodes, then the pattern is responsible for removing them. Relationships are destroyed automatically when one or both of the nodes they relate is destroyed. You can also consider using Group removal functions or `model.destroy`. Relationships are not destroyed when the pattern that created them is deleted. If you destroy a node to remove relationships, you should rescan the infrastructure concerned in order to recreate the removed nodes and relationships. If you are using CMDB synchronization, you should resynchronize all of the hosts associated with these nodes to ensure that the CMDB is synchronized.

**Unique relationship functions**

It is often useful to declare that a node is related to nothing but a specific node (or nodes). For example, if a Host is in a particular Location, it is not in any other Location. For each relationship defined in the taxonomy, a function is defined in the `model.uniquerel` scope. It takes the same form as the equivalent `model.rel` function, but its behaviour is different:

```python
model.uniquerel.relationship(source_role := source, dest_role := dest, att
```

- If a relationship already exists between the source and destination, its attributes are updated.
- If a relationship does not exist from the source to the destination it is created.
- All other matching relationships from the source to other destinations are destroyed.

⚠️ **Parameter order**

Unlike `model.rel` functions, in `model.uniquerel`, the order the roles is specified in is important. The first listed role belongs to the node that has the unique relationship; the node with the second listed role is permitted to have more than one matching relationship. i.e. a Host can be in only one Location, but a Location can have many Hosts in it.
A relationship matches for destruction if it has the same relationship kind, same source and
destination roles, and has attributes that match the values of the specified attributes.

For example, to relate a Host to a location:

```plaintext
place := // a Location nodemodel.uniquerel.Location(ElementInLocation := host, Location := place);
```

The use of `uniquerel` ensures that the Host nodes are only ever related to one Location.

As a more advanced example, attribute matching can be used to relate to both a city location and
an office location:

```plaintext
// City patternplace := // a Location node for the citymodel.uniquerel.Location(ElementInLocation := host, Location := place, type := "city");
// Office patternplace := // a Location node for the officemodel.uniquerel.Location(ElementInLocation := host, Location := place, type := "office");
```

With those patterns, each Host will now be related to two Location nodes. The use of `uniquerel`
in the first pattern will only destroy Location relationships with a `type` attribute of "city", and the
second one will only destroy relationships with a `type` of "office".

Like `model.rel` functions, lists of nodes can be provided. A list in the source role simply has the
same effect as calling the function multiple times, once with each source node. A list in the
destination role, however, causes the source to be related to all of the destination nodes, and
destroys any relationships to other nodes.

⚠️ **Use of `model.uniquerel.Inference` should be avoided**

You should avoid using the `model.uniquerel.inference` function. The inference
system expects specific attributes on the Inference relationship. If these attributes are
missing there might be a traceback whenever the system attempts to update inference
information. A warning is logged when you upload a pattern using the `model.uniquerel.inference` function.

The functions in `model.uniquerel` were introduced in TPL 1.2.

**Model containment functions**

To support construction of Software Instance, Business Application Instance and virtual Host
models, two containment specification functions are provided:

- `model.addContainment`
- `model.setContainment`

**Model maintenance functions**

The following functions also exist in the `model` scope to remove nodes or their attributes:
- model.destroy
- model.withdraw

Group removal functions
When patterns construct complicated structures, such as DatabaseDetail nodes when doing deep discovery of databases, removal must be performed by the pattern. The following functions simplify deletion of these complex structures:

- model.setRemovalGroup
- model.anchorRemovalGroup
- model.suppressRemovalGroup

The system expects nodes to be in one removal group only. Where a node is in multiple removal groups, the behavior is undefined.

Model traversal functions
The following convenience functions in the model scope aid traversals across the model. They take positional arguments:

- model.host
- model.hosts
- model.findPackages

model.addContainment( container , containees )
Adds the containees to the container by creating suitable relationships between the nodes. containees can be a single node, or a list of nodes, for example: [node1, node2].

If the containees are Software Instances or BusinessApplications instances, SoftwareContainment relationships are created; if the containees are Hosts, HostContainment relationships are created. For software containment, appropriate HostedSoftware relationships are also created to link the container to the correct Hosts according to the contained nodes.

model.setContainment( container , containees )
Equivalent to addContainment, except that at the end of the pattern body, any relationships to contained nodes that have not been confirmed by setContainment or addContainment calls are removed.

model.destroy( node )
Destroy the specified node or relationship in the model. (Not usually used in pattern bodies, but in removal sections.)

model.withdraw( node , attribute )
Removes the named attribute from the node.

model.setRemovalGroup( node , [ name ])


Add the specified node or nodes to a named removal group. The purpose of this function is to identify a group of nodes. If a group name is not specified the default name group is used. Each identity is scoped to an individual pattern. Those nodes must be created and managed solely by the pattern making the call. Adding nodes managed by other patterns will have undefined behavior. The function can be called multiple times within the pattern.

On completion of the pattern all nodes created or confirmed as belonging to a group in the previous execution of the pattern are examined. If a node does not appear in the same group in the new run it is deleted.

By default the system associates the group nodes with the trigger node. This means that the trigger node must be an inferred node or it will differ with every invocation of the pattern. This can be overcome by specifying an inferred anchor node for a named group.

You can prevent a removal group being removed using the model.suppressRemovalGroup function.

For example, the following code creates a group named dbs. On the first run this contains all the discovered databases. On the second run, if any databases are not seen then they are automatically removed upon the completion of the pattern.

```model
for row in dbs do  db := model.DatabaseDetail(key := row.key, name := row.
```

Changing the name of a removal group

In the course of writing a pattern you might choose to change the name of a removal group. When you upload the new pattern with the changed removal group name, and the pattern runs, the original removal group and the nodes it contains are removed. The removal group with the new name is created, but does not have the same nodes added as they have been removed. On subsequent scans the nodes are recreated and are added to the new group when the pattern runs.

```model
model.anchorRemovalGroup( node , [ name ])
```

Specify an anchor node for a named removal group. If a group name is not specified the default name group is used. By default the system associates the group nodes with the trigger node. This means that the trigger node must be an inferred node or it will differ with every invocation of the pattern. If a pattern triggers on a DDD node, you can anchor the group to a specified inferred node.

When you use an anchor in a pattern you must ensure that the same anchor is used on each subsequent invocation of that pattern. If you do not do this, no nodes in the removal group will be deleted.

For example, a pattern triggers on a DiscoveredProcess and creates an SI called dbsi. The following code anchors the group named dbs to the inferred node, the SI called dbsi.

```model
model.anchorRemovalGroup(dbsi, "dbs")
model.suppressRemovalGroup([ name ])
```
Suppress removal of the named removal group. If a group name is not specified the default name group is used. Removal groups aid removal of complex structures such as those created by deep discovery of databases (see model.setRemovalGroup for more information). In certain circumstances you might want to prevent removal of the removal group.

For example, the following code sets a removal group.

```java
for row in dbs do  db := model.DatabaseDetail(key := row.key, name := row.name);model.setRemovalGroup(db, "dbs");end for;
```

If the pattern fails to recover details from the database, all the database details would be removed. This can be prevented using the following code in the case for example, where the credentials are locked out:

```java
model.suppressRemovalGroup("dbs");
```

model.host( node )
Returns the Host node corresponding to the given node. The given node can be any directly discovered data node, or a Software Instance or Business Application Instance. If more than one Host is related to the given node (for example a Business Application Instance spread across multiple Hosts), an arbitrary one of the Hosts is returned.

⚠️ **Name clash**

The model.host function, with a lower-case "h" has an unfortunate clash with model.Host with an upper-case "H", which creates or updates an existing Host node. Be sure to use model.host with correct capitalisation. Since patterns are not normally responsible for maintaining Host nodes, the system issues a warning if model.Host is used.

model.hosts( node )
Returns a list of all the Host nodes corresponding to the given node. As with model.host, the given node can be any directly discovered data node or a Software Instance or Business Application Instance.

model.findPackages( node , regexes )
Traverses from the node, which must be a Host or a directly discovered data node, and returns a set of all Package nodes that have names matching the provided list of regular expressions. This is commonly required when versioning products based on installed packages. Returns an empty set if the node was not of a suitable kind, or no matching packages were found.

model.addDisplayAttribute(node, value)
Adds a named attribute, or a list of named attributes to the additional attributes displayed in a node view. Added attributes can be removed using model.removeDisplayAttribute.

Example usage is shown in adding custom attributes.
Removes a named attribute, or a list of named attributes from the additional attributes displayed in a node view. Additional attributes are added using the `model.addDisplayAttribute` function.

Example usage is shown in adding custom attributes.

**Related functions**

The following are related functions. Each takes a single positional argument:

- `related.host` - (8.3 and later)
- `related.detailContainer` - (8.3 and later)

**related.detailContainer( node )**

Returns the Software Component, Software Instance, or Business Application Instance node containing to the given node. The given node can be a Detail or DatabaseDetail. If no single container node can be found None is returned.

**related.host( node )**

Returns the Host node corresponding to the given node. The given node can be any directly discovered data node, or a Software Instance or Business Application Instance. If more than one Host is related to the given node (for example a Business Application Instance spread across multiple Hosts), an arbitrary one of the Hosts is returned.

**Email functions**

The `mail` scope contains a function for sending email:

- `mail.send` — (7.1 and later)

**mail.send( recipients , subject , message )** — (7.1 and later)

Sends an email. `recipients` is a single string containing an email address, or a list of email address strings; `subject` is the subject line to use in the email; `message` is the message contents.

BMC Atrium Discovery must be correctly configured with an SMTP server for email sending to succeed. Sending a large number of emails could potentially impact performance and should be used with caution.

**Time functions**

The following functions manipulate dates and times. They were added in TPL 1.2.

There are two ways that dates and times are represented in BMC Atrium Discovery. Within TPL, dates and times are represented as a specific time type, that can be manipulated and stored as times. Other parts of the system, including the time-related search functions, manage times as large integers, representing the number of 100 nanosecond intervals since midnight UTC on the 15 October 1582 (the date of the introduction of the Gregorian calendar). This is known as "ticks" or "DCE format"; there are 10 million ticks per second. In both cases, the times are stored in UTC.
TPL contains functions for converting between the two formats. Aside from the conversion functions, all TPL operations use the time type, not the integer ticks format.

⚠️ **Type mixing**

Be careful to understand what kind of time value you are manipulating, either the internal TPL time format or the integer "ticks" format. Mixing the two can lead to errors.

- `time.current`
- `time.delta`
- `time.parseLocal`
- `time.parseUTC`
- `time.formatLocal`
- `time.formatUTC`
- `time.toTicks`
- `time.fromTicks`
- `time.deltaFromTicks`

**time.current()**

Returns the current time.

**time.delta( days , hours , minutes , seconds )**

Creates a time delta that can be added to or subtracted from a time. It accepts keyword parameters, for example:

```plaintext
delay := time.delta(hours := 5, minutes := 15);
now := time.current();
later := now + delay;
```

Time deltas cannot be stored in the datastore. Attempts to do so will result in a run-time error.

**time.parseLocal( string )**

The time is converted according to the appliance's time zone and daylight saving setting.

**time.parseUTC( string )**

The time is assumed to already be UTC, and is not adjusted for timezones or daylight saving time.

**time.formatLocal( datetime [ , format ])**

Formats a time into a human-readable string. With no format specification, the default BMC Atrium Discovery time representation is used. If given, the format specification is a string containing special tokens that expand to formatted parts of the date and time. The format is specified as in Python's `time.strftime` function. As with the parsing functions, formatLocal converts the time into the appliance's timezone and daylight savings setting; formatUTC performs no timezone conversion.

**time.formatUTC( datetime [ , format ])**
Formats a time into a human-readable string. With no format specification, the default BMC Atrium Discovery time representation is used. If given, the format specification is a string containing special tokens that expand to formatted parts of the date and time. The format is specified as in Python's time.strftime function. As with the parsing functions, formatLocal converts the time into the appliance's timezone and daylight savings setting; formatUTC performs no timezone conversion.

time.toTicks( datetime )
Converts a time or time delta into the DCE ticks format.
time.fromTicks( ticks )
Converts DCE ticks into a time.
time.deltaFromTicks( ticks )
Converts DCE ticks into a time delta.

Inference functions
The inference scope contains functions to permit explicit management of inference relationships that represent data provenance in the model. The pattern language automatically maintains inferences in most situations, so these functions are only required in rare complex circumstances.

- inference.associate
- inference.contributor
- inference.primary
- inference.relation
- inference.withdrawal
- inference.destruction

inference.associate( inferred_node , associate )
Create associate inference relationship(s) from the specified node(s) to the inferred node.
inference.contributor( inferred_node , contributor , contributes )
Create contributor inference relationship(s) from the specified node(s) to the inferred node, for attribute names specified in the contributes list.
inference.primary( inferred_node , primary )
Create primary inference relationship(s) from the specified node(s) to the inferred node.
inference.relation( inferred_relationship , source )
Create relation inference relationship(s) from the specified node(s) to the inferred relationship.
inference.withdrawal( inferred_node , evidence , withdrawn )
Create withdrawal inference relationship(s) from the specified node(s) to the inferred node, indicating the withdrawal of the withdrawn attribute name.
inference.destruction( destroyed_node , source )
When destroying a node, indicate that the source node was responsible for its destruction.

System functions
The system scope contains a single function to retrieve the values of BMC Atrium Discovery system settings.
- **system.getOption**

**system.getOption(option_name)**
Takes the name of a BMC Atrium Discovery system option and returns the value.

Where:

- **option_name** – the name of the BMC Atrium Discovery system option to be returned.

If an invalid option name is supplied, the call fails and returns nothing.

**Search expressions**

The full search syntax is supported with a small number of minor modifications for consistency with the rest of the pattern language. The modifications to the search syntax are as follows:

- Searches have syntax similar to a function call where the expression after the `search` keyword must be in parentheses.
- All search keywords must be in lower case.
- Regular expressions must be matched with the keyword `matches` rather than `like`.
- Identifiers clashing with keywords must be prefixed with `$` instead of `~`.
- Post-processing functions must be specified with the single keyword `processwith`, rather than two separate words `process with`.

Outside of the pattern language, the search service is backwards-compatible and supports `like` as a synonym for `matches`, `process with` as a synonym for `processwith`, and accepts keyword escaping with `~` as well as `$`.

The result of a search expression depends on what kind of `show` clause it has. If the search has no `show` clause, the result is a node set; if the search has a `show` clause with one single attribute shown, the result is a list of values; if the search has a `show` clause with multiple attributes shown, the result is a list of lists of values. In searches with a `processwith` clause, the result is either a node set or a list of lists of values, depending on the definition of the post-processing function used.

A search expression has an optional `in` clause which triggers a refine search, as opposed to a search of the whole datastore. The expression given to `in` must be a single node or a node set.

This example constructs a set of all processes discovered at the same time as the given process, then refines the set:

```
process := // acquire DiscoveredProcess node from somewhere
all_procs := search(in process
  traverse Member:List:List:ProcessList
  traverse List:List:Member:DiscoveredProcess);
required_procs := search(in all_procs where cmd matches "httpd");
```
Attribute names used in search expressions (except the `in` clause) are scoped to the search expression, not to the body of the pattern. To access variables in the pattern body, `%` symbols must be used as in string interpolation:

```plaintext
instance := // instance name from somewhere
db_nodes := search(SoftwareInstance
where kind = "Sybase" and
instance = `%instance%`);
```

**Search post-processing functions**

A number of search `processwith` functions are useful for writing patterns that relate items to each other based on observed network connection information. For example, the following example (TPL 1.1 and later) finds all the Software Instances that are communicating with a particular Software Instance:

```plaintext
si := model.SoftwareInstance(key := key details, ...);
communicating := search(in si processwith communicatingSIs)
```

**User defined functions**

TPL 1.5 introduces user defined functions. These can be imported from other modules and reused.

**Defining functions**

User defined functions are specified in `definitions` blocks. In order to specify a function the `type` parameter within the `definitions` block must be set to the value `function` or omitted.

Each `define` within the `definitions` block specifies a function. A function definition consists of:

- A pair of parentheses containing an optional list of identifiers. The identifiers represent the number of parameters specified in a call to the function and the identifier used to reference each parameter.
- An optional return list specified by following the close parenthesis with `->` and a list of identifiers, one for each return value.
- A description string
- A list of statements

The following example shows a simple example function:

```plaintext
tpl 1.5 module example;
definitions functions 1.0'User defined functions' type := function; // Or
define reverse(x, y) -> a, b'swap two values' return y, x;end define;
```

You can return a comma separated list of values using the `return` statement. This list must have the same number of values as specified by the function definition. The return value identifiers are purely symbolic and not referenced anywhere within the function call or definition. For example:

```plaintext
define reverse(x, y) -> a, b'swap two values' return y, x;end define;
```
You can specify a default value for function parameters. Any parameter following the first that has a default value must also have a default value. In the following example, calling `add` with a single parameter returns `x+2+3`.

```define add(x, y:=2, z:=3) -> a'add values, using defaults where parameters are not passed'    return x+y+z;end define;```

Here are some more simple examples of functions.

```define rnd() -> y'functions.rnd is a function which returns a value.'    return 16;end define;```

```define update(node, key, value)'''functions.update is a function which takes three parameters and does not return a value.''' node[key] := value;end define;```

Using functions

The following pattern contains calls to functions defined in the same TPL module. The functions are the examples given in the preceding section.

```pattern function_caller 1.0'Pattern which calls functions.'  overview    tags example;end overview;triggers    on process ...     // call reverse which returns                                         // two values, 2 and 1.  end body;end pattern;```

Recursion

Recursive and mutually recursive functions are not supported. Attempts to use recursion will be diagnosed, preventing the pattern module from being activated.

Functions calling functions

If you define a function which calls another function, the function being called `must` be defined before the function doing the calling, otherwise the pattern will not activate.

Removal

Patterns that are triggered on directly discovered data can specify specific rules for removing nodes and relationships previously created by the same pattern. If there are no removal specifications, removal for nodes created by such patterns is based on aging. Removal for nodes created by patterns triggered from other data (such as other inferred nodes) is based on existence of the originally triggering nodes.

Removal blocks take the form:

```removal on name := node_kind removal_condition [ where condition ];// removal body...end removal```

An example is:

```removal on si := SoftwareInstance aged;// This SoftwareInstance has a fl```
If a pattern creates multiple nodes, the removal block is executed for each of them when the removal condition is met. More than one removal block can be specified to support patterns that create more than one kind of node, and to support different blocks where the node kind is the same, but the conditions are different.

Once a pattern has at least one removal block, the normal automatic aging of all the inferred nodes the pattern creates is disabled. Explicit removal rules must be provided for all the created nodes.

Removal conditions
Removal conditions specify when the removal block will be executed. Two removal conditions are currently supported: aged and unconfirmed.

aged
aged caters for the case where something is about to be aged out, but you want to do additional checks to see if the system should actually delete it. In other words, you want to override the default aging behavior in order to make the system not delete something that it otherwise would do.

An example is where you have an SI modeled on a process that only runs rarely: it's likely that at some point the SI will be about to age out because that process hasn't been seen in a while, but what's actually happened is that the process has run, just not at the time that the host was discovered.
If aged is specified, the removal block is executed at the time the node in question would normally be removed due to aging; that is, it has not been confirmed for a while according to the aging parameters.

Because only root nodes and first-order SIs are removed through aging, it is important that you maintain your other nodes created in patterns. These nodes are not maintained automatically by BMC Atrium Discovery. For more information about data aging settings and the guidelines around modifying them, see Configuring model maintenance settings.

unconfirmed
unconfirmed caters for the case where you want to remove something as soon as possible after it disappears from your environment. In other words, you want to override the default aging behavior in order to remove something before the system would have done.

An example is where you have a policy engine checking your CMDB to ensure that a given computer system is running a given piece of software. Thus, if that software is not seen when you discover the host then that indicates a policy violation, and you need to instantly delete the software instance from the data store (and thus from the CMDB) so that your policy engine can notice its absence.

If unconfirmed is specified, the removal block executes every time a scan of the relevant host is performed without the node in question being confirmed.
The where clause

The optional where clause can be used to only trigger execution of the removal block when the inferred node matches the criteria. The conditions take the same form as those in the trigger block (see trigger conditions).

Pattern Configuration

It is sometimes useful to permit the end-user to modify certain characteristics of a pattern, such as whether to enable features that might be costly, or to change a set of default paths or package names. This is enabled with pattern configuration blocks. Pattern configuration is new in TPL 1.2.

ℹ️ Changes in the pattern configuration block revision number may reset the pattern configuration changes to the default values

If you install a Technology Knowledge Update (TKU) which contains an update to a pattern configuration block that requires its major revision number to be incremented, any end-user configuration changes to that pattern (made through the pattern configuration block) are lost and the default pattern values are restored.

Configuration block definition

configuration blocks are top-level elements. Like other elements, they have a name and a version number. Pattern modules can import configuration blocks from other modules, using the usual version numbering conventions.

The form of a configuration block is:

configuration name version "" Description of configuration block to show in UI

...end configuration;

Configuration items are given a human-readable description to be shown in the user interface, an identifier to use in patterns, and a default value. The default value is the value used if the configuration item has not been changed by the user. It also determines the type of the configuration item — the user interface will only allow the user to change the setting to a value with the same type. The valid types for configuration items are text string, integer, boolean, list of strings, and list of integers.

This example shows a number of configuration settings:

configuration MyConfig 1.0"" An example configuration block."" "" A bool
"Command to run"  command := "sudo rm -rf /";
"Port for the service"  port := 12345;
"Paths to look in"  paths := [ "/usr/bin", "/usr/local/bin" ];
end configuration;

Using configuration settings

Configuration settings are used in patterns by giving the scoped name of the configuration block, followed by the configuration setting name:

if MyConfig.a_flag then  result := discovery.runCommand(host, MyConfig.command)

Types of default empty lists

The types of configurations items can usually be inferred from the type of the default value. However, if the default value of a configuration item is an empty list, the system cannot know whether it is a list of text strings or a list of integers. To tell the system the type of such a configuration item, an example of the type should be provided in parentheses after the empty list:

"This is a list of strings that's empty by default"empty_string_list := [ ()

"This is a list of integers that's empty by default"empty_number_list := [ 0];

Static Tables

Tables provide simple look-up tables that can be used by functions in pattern bodies. They have no active behaviour. Tables must be declared in module scope, not inside pattern declarations. Declarations have the following form:

table  name  version  key1 -> value1;key2 -> value2;
    ...end table;

Keys are literal values or expressions involving only literals; values are comma-separated sequences of literal values, or expressions involving only literals.

Tables can have a default value, specified with a key of default:

table  example_table  1.0"one" -> "two";"three" -> "four";default -> "five";

Tables are accessed in pattern bodies using square brackets:

table_key := "one";
value := example_table[table_key];

For details of dynamic table use from TPL 1.2 onwards, see Table functions
Enumerations

Enumeration maps a list of names to either a string or integer value. Within a module the enumeration has a similar form to other top level blocks. The following code defines an enumeration `enum_name` with three values `identifier_1`, `identifier_2` and `identifier_3`.

```
tpl 1.4 module example;
enumeration enum_name 1.0'Enum description'  identifier_0,
               identifier_1,
               identifier_2end enumeration;
```

If no values are given to the names then the first value will be 0. Each subsequent value will be one larger than the previous value. So in this example `identifier_0` is 0, `identifier_1` is 1 and `identifier_2` is 2. The default values can be overridden:

```
tpl 1.4 module example;
enumeration enum_name 1.0'Enum description'  car -> 5,
               boat,
               plane -> 8,
               tankend enumeration;
```

Using the `->` operator allows the value to be replaced. In this example, `car` is 5, `boat` is 6, `plane` is 8 and `tank` is 9. It is legal to have multiple names mapping to the same value though the names themselves must be unique. Alternatively an enumeration can map to string values. This is done by giving one of the values a mapping to a string. Any unmapped value will take the string representation of its name. Given

```
tpl 1.4 module example;
enumeration enum_name 1.0'Enum description'  car,
               boat -> "water",
               planeend enumeration;
```

`car` is "car", `boat` is "water" and `plane` is "plane". Strings and numbers cannot be mixed within a single enumeration.

Using enumerations

Enumeration values can be used in metadata or in an assignment within a pattern. Metadata has been extended so it now also supports numbers.

```
tpl 1.4 module example;
enumeration vehicle 1.0'Enum description'  car,
               boat -> "water",
               planeend enumeration;

pattern example 1.0'Test pattern'  overview  tags example;end overview;t
The pattern assigns the variable a the value "water". Enumeration values cannot be used within expressions.

**Identify**

Identify tables are active tables used to annotate matching nodes with particular values. As with non-active Tables, they must be declared at module scope, not inside patterns.

They take the form:

```
identify name version
  [metadata
    metadata_entries
  end metadata;]
  tags tag1, tag2, ... ;
  node_kind matchattribute [ , . ] -> _set_attribute [ , ... ];
  key1 -> value1;
  key2 -> value2;
  ...
end identify;
```

The identify table is triggered whenever a node is created with suitable attributes for the match attributes. Like patterns, identify tables must declare one or more tags.

Upon triggering, it sets the set attributes on the triggering node. This is used for simply identifying processes, for example:

```
identify common_unix_commands 1.0
  tags example;
  DiscoveredProcess cmd -> simple_identity;
  unix_cmd "ls" -> "Unix directory listing command";
  unix_cmd "mv" -> "Unix move command";
  unix_cmd "cp" -> "Unix copy command";
  ...
end identify;
```

If more than one regular expression in an identify table matches a particular node, an arbitrary one will "win" and set the corresponding value.

**Definitions**

Additional functions to call in patterns are described in definitions blocks. In TPL 1.5, definitions blocks are used for User defined functions and for data integration with SQL databases.

The basic form of a data integration definitions block is:
```plaintext
definitions name version  """Description of definitions block"""
  type := definitions_type; other_setting := setting_value;
  ...
  define function_name    """Function description""""  parameters := parameter
  ...
end definitions;
```

All definitions blocks must have a type specification. Other settings might be required by certain definitions types. The definitions block contains one or more define blocks, describing the functions being defined. All functions that take parameters must have a parameters specification, listing the names of the parameters to the function. (If a function takes no parameters, the parameters specification can be missed out.) Most definitions types also require other settings in the define blocks.

Once defined, functions can be called in patterns with the usual function calling syntax. Definitions can be imported from one module into another, with the usual versioning scheme.

The supported definitions types are as follows:

- SQL database integration
- SQL database discovery

**SQL database integration**

Centralised databases — asset databases, for example — are accessed with `sql_integration` definitions. The definitions block must contain a name setting, which corresponds to the Integration Point in the user interface. Each define block must contain a query setting that contains the SQL query to perform. Parameters are inserted into the query with the usual TPL `%` interpolation. e.g.

```plaintext
definitions AssetDatabaseDetails 1.0"""Queries to obtain information from

  name := "Example asset database";
  define getLocationOfHost"""Return details of the location for the given host

    query := """select name, address from hosts, locations where
             hosts.id_location = locations.id and hostname = %hostname%

end definitions;
```
The defined function is used in a pattern as the function AssetDatabaseDetails.getLocationOfHost. It is called with a hostname parameter, followed by a connection parameter which refers to a connection created within the Integration Point. It returns a list of nodes, one for each row of output from the query. If the query fails in some way, the function returns none. Each row node has attributes named after the columns selected in the SQL query. So, for example, to use the query defined above:

```python
location_rows := AssetDatabaseDetails.getLocationOfHost(hostname := host.hostname,
connection := $$connection_name$$;

for row in location_rows do  // Create / update Location node for this loc

    address := row.address,
    key := "%row.name%.%row.address%";// Relate the I
```

### SQL database discovery

Database integration is used when there is a single, centralised database to be accessed. Database discovery, on the other hand, is used to perform queries on databases that have been discovered running on discovery targets. It is analogous to running commands on discovered hosts. SQL discovery is specified with sql_discovery definitions. The definitions block must contain a group setting, which corresponds to a Database Credential in the user interface.

Like sql_integration definitions, the define blocks in sql_discovery ones require a query setting, which gives the SQL query to be used.

When calling an sql_discovery function, the target for the discovery must be specified via additional parameters. The functions take an endpoint parameter, which is a Host node or DDD node that identifies the target (like the target in built-in Discovery Functions). Some database types require additional parameters to identify the target database, and most have a number of optional parameters. See the Driver Documentation for details.

Additionally databases might be listening only an a specific interface. Default behaviour when a Host or DDD node is passed in the endpoint parameter is to attempt the connection on the IP currently used for discovery. If this is not the specific interface that the database is listening on this connection will fail. To avoid this an additional address parameter can be passed. This should contain a string with the desired IP address, if this address is one of the current interfaces on the Host this will be used instead of the default behaviour.

Here is a code snippet making a call into MySQL

```python
// Use MySQL provider to get database info if mysql.bind_address then
    all_dbs := MySQLDetails.showDatabases(endpoint := host,
```
Node names, types and keys

The majority of nodes created by patterns are Software Instances and Business Application Instances. Both node kinds have attributes name and type, and, in common with all nodes maintained by patterns, must have a key. This appendix discusses how to set those attributes in a number of scenarios. In the following scenarios, the node kind is consistently referred to as a Software Instance (SI), but equivalent suggestions apply to Business Application Instances (BAIs).

The name attribute is intended to be a human-readable summary of the identity of the node - it need not be globally unique, but it should give the reader a good idea about which instance it is.

SI on single host, uniquely identified and versioned

**Scenario 1** — The SI is running on a single host. The pattern can determine a unique identity, and determine the version for each SI:

<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance</td>
<td>The unique instance name of the SI, for example, &quot;TRADING_DB&quot; or <a href="http://www.example.com">www.example.com</a>.</td>
</tr>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% version %version% called %instance% running on %host.name%&quot;, for example, &quot;Oracle Database version 10g called TRADING_DB running on londb01&quot;. As appropriate, the word &quot;called&quot; should be replaced with similar words such as &quot;identified by&quot; or &quot;using&quot;. The version number used should either be the product version or the full version, whichever is most appropriate for the product.</td>
</tr>
<tr>
<td>key</td>
<td>A string of the form &quot;%instance%/%type%/%host.key%&quot;</td>
</tr>
</tbody>
</table>

SI on single host, uniquely identified but not versioned

**Scenario 2** — The SI is running on a single host. The pattern can determine a unique identity, but cannot determine the version for each SI:

<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance</td>
<td>The unique instance name of the SI, for example, &quot;TRADING_DB&quot; or &quot;counterexamples&quot;.</td>
</tr>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% called %instance% running on %host.name%&quot;, for example, &quot;Oracle Database version 10g called TRADING_DB running on londb01&quot;. As appropriate, the word &quot;called&quot; should be replaced with similar words such as &quot;identified by&quot; or &quot;using&quot;.</td>
</tr>
<tr>
<td>key</td>
<td>A string of the form &quot;%instance%/%type%/%host.key%&quot;</td>
</tr>
</tbody>
</table>

SI versioned but not uniquely identified

**Scenario 3** — Cannot determine a unique name for each SI, but can find version
<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% version %version% running on %host.name%&quot;.</td>
</tr>
<tr>
<td>key</td>
<td>Not set.</td>
</tr>
<tr>
<td>key_group</td>
<td>version. The system will create a suitable key embedding the version given as the key group, the type and the host's key</td>
</tr>
</tbody>
</table>

**SI not versioned or uniquely identified**

**Scenario 4 — No unique name or version available**

<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% running on %host.name%&quot;.</td>
</tr>
<tr>
<td>key</td>
<td>Not set.</td>
</tr>
<tr>
<td>key_group</td>
<td>Not set, or set to the empty string. The system will create a suitable key embedding the type and the host's key</td>
</tr>
</tbody>
</table>

**Product with only one instance on a host**

**Scenario 5 — A product that can fundamentally only have one instance running on a host**

<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% running on %host.name%&quot;.</td>
</tr>
<tr>
<td>key</td>
<td>A string of the form &quot;%type%/%host.key%&quot;.</td>
</tr>
</tbody>
</table>

**An SI or BAI spanning multiple hosts**

**Scenario 6 — A Software Instance or Business Application Instance spanning multiple hosts**

<table>
<thead>
<tr>
<th>Type</th>
<th>The name of the product, for example, &quot;Oracle Database&quot; or &quot;Apache Web Server&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance</td>
<td>In this case, the SI / BAI is composed of other SIs / BAIs. To group them together, there must be some identifying characteristic for the group. In cases where there is one global instance of the application, the instance would not be set.</td>
</tr>
<tr>
<td>name</td>
<td>A string of the form &quot;%type% version %version% called %instance%&quot;. As appropriate, the word &quot;called&quot; should be replaced with similar words such as &quot;identified by&quot; or &quot;using&quot;. If there is one global instance, the instance information should be left out. The version number used should either be the product version or the full version, whichever is most appropriate for the product, or absent altogether if no version information is available.</td>
</tr>
<tr>
<td>key</td>
<td>A string of the form &quot;%instance%/%type%&quot;</td>
</tr>
</tbody>
</table>

**Syncmapping block**

The mappings performed during CMDB synchronization are specified in syncmapping blocks in TPL. A syncmapping is similar to a pattern, but it triggers from a queued synchronization action, rather than from data being updated in the data store.
The form of a `syncmapping` is:

```plaintext
syncmapping name version
description
overview
overview_entries
end overview;
[constants
constant_definitions
end constants;]
mapping mapping_source as source_name
mapping_definitions
end mapping;
body
body_details
end body;
end syncmapping;
```

As with pattern blocks, the name, version, and description are mandatory.

Pattern templates are provided to help you create your own syncmappings.

**Overview section**

The `overview` is required. It contains information about the pattern and the entities it creates. It must contain a `tags` entry, and can have an optional `datamodel` entry, as described in Data models.

The overview section can also have an optional `overrides` entry, which enables you to change the behavior of an existing mapping and to preserve those changes on subsequent TKU updates to the overridden mapping.

Here is an excerpt from an overriding mapping:

```plaintext
...from CMDB.Host_ComputerSystem import Host_ComputerSystem 1.2; from CMDB.

mapping from Host_ComputerSystem.host as host_traverse Host:HostedSoftware:

traverse Element:Maintainer:Pattern:Pattern as si_pattern end traverse;
...
```

---

**Internal Reference:**

CMDB, Core_Mapping; 0, 1, 2, 3, 4, 5, 6; SoftwareInstance_SoftwareServer;  // The overridden pattern is imported above;

---

**Internal Reference:**

Host_ComputerSystem.host

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem

**Internal Reference:**

Host:HostedSoftware

---

**Internal Reference:**

Host_ComputerSystem
The mapping referred to by the `overrides` keyword must be imported. All definitions from the original mapping section such as node kinds, traversals and names must be preserved. It is possible to add additional traversals and names in the overriding mapping.

### Mapping section

The `mapping` section declares the starting point for the mapping, the structure of source data retrieved from the Discovery model, and the target CIs created in the CMDB model. It does not describe _how_ the source data is transformed to the target model — that is performed in the `body` section.

### Mapping source

Each mapping is either a `root` mapping, meaning that it is invoked by the synchronization of a single root node with the corresponding kind, or an `extension` mapping, meaning that it extends another mapping at a suitable point.

Root mappings have a `mapping` declaration using the `on` keyword:

```
mapping on node_kind as name mapping content...end mapping
```

For example, this specifies the root mapping for Host nodes:

```
mapping on Host as host_node
```

Extension mappings have a `mapping` declaration using the `from` keyword:

```
mapping from source_scoped_name as name
```

For example:

```
mapping from ExampleMapping.host_node as host_node
```

The `source_scoped_name` is the name of a source mapping variable from another `mapping` block, either the source name specified in the `mapping` declaration, or a traversal name as described below.

### Mapping traversals

The source subgraph is declared using `traverse` clauses inside the `mapping` with a syntax similar to traversals in search expressions:

```
traverse traversal_specification as name traversal contents...end traverse
```

The `name` defined by the traversal can only be used in a `for each` expression; it cannot be used in any other context.
Where clauses
The initial source node and the results of traversals can be filtered with *where* clauses, specified before the *as* token. *where* clauses in *mapping* blocks use the same subset of search *where* clauses as *trigger conditions* in *pattern* blocks.

```
mapping on node_kind where condition as name
mapping from sourceScopedName
```

Target CI declarations
As the subgraph is processed in the *body*, target CIs are specified. The *mapping* block contains declarations of the CIs that are mapped, in the form:

```
name -> CI_class;
```

For example:

```
computersystem -> BMC_ComputerSystem;
```

Targets are specified within the traversal structure. For example, part of the mapping of virtual machines is as follows:

```
mapping from Host_ComputerSystem.host where virtual defined as host
traverse
vse -> BMC_VirtualSystemEnabler;
traverse RunningSoftware:HostedSoftware:Host:Host as containing_host
containing_cs -> BMC_ComputerSystem;
end traverse;
end traverse;
end traverse;
mapping
```

Grouping
In some circumstances, a number of nodes in the Discovery model must be grouped together to construct a single CI in the CMDB model. This is declared in the *mapping* with a *group* block. The form of a group block is

```
traverse traversal_specification as traversal_name group group_name
```

The declaration indicates that nodes from the containing *traversal* will be grouped together (according to rules specified in a *group* block in the *body*), and then the group will be expanded to the individual group members. The *expand* is not required if there is no need to process the individual nodes within the group.

Syncmapping body
The *body* of a *syncmapping* is responsible for implementing the mapping described in the *mapping* block. The majority of language features and functions available in *pattern body* blocks are permitted, except that functions in the following namespaces are not available since they are only appropriate for patterns that perform discovery and construct the Discovery model.
• discovery
• inference
• mail
• model

Additionally, user-defined functions are not supported.

**Body execution**

The body of a syncmapping is executed at a time that depends upon the mapping source definition.

The body of a root mapping (specified with mapping on) is executed at the time the root node is scheduled for synchronization.

The body of an extension mapping (specified with mapping from) that extends the source node of another mapping is executed when the body of the extended mapping completes.

The body of an extension mapping that extends a traversed-to node of another mapping is executed each time the associated for each loop (see below) completes.

When a node in the Discovery data store is marked as destroyed, only the root mapping's body is executed, and the target root CI is scheduled for deletion in the CMDB. When the delete is synchronized with the CMDB, the root CI and all the related CIs previously created by the mapping are deleted. For best performance during deletion, root mappings should not perform any traversals or other time-consuming activities.

**CI definition**

CIs and relationships in the CMDB are specified with functions in the sync namespace similar to those in the model namespace used within pattern blocks.

Any CMDB class can be specified with a function call of the form

```
value := sync.BMC_ClassName(key := some_key, attributes...);
```

Any class name can be specified. Specifying a class that is not defined in the CMDB results in a run-time error. The key attribute must be set, and is used to populate the ADDMIntegrationID attribute in the CMDB. Any other attribute name can be set; attributes that are not defined in the CI class are ignored.

The result of the function must be assigned to a target CI name specified in the mapping block. The class specified in the function must be the same as the one specified in the mapping or a subclass of it.
CI class namespaces
CMDB classes are assumed to be in the `BMC.CORE` namespace. To refer to a class in a different namespace, provide a `namespace` parameter to the function call:

```plaintext
value := sync.BMC_ClassName(key := some_key, namespace := "My.NameSpace",
```

The namespace must be a literal string — it cannot be constructed at runtime.

Shared CIs
The subgraph of data in the Discovery model is transformed into a subgraph of CIs in the target CMDB model. Most of the CIs belong to a single target subgraph, but some are shared by more than one subgraph. An example is the `BMC_IPConnectivitySubnet` CI that is shared by all the computers on a particular subnet. For deletion to work correctly, the system must know that such CIs are shared. This is achieved by calling the function in the `sync.shared` namespace:

```plaintext
value := sync.shared.BMC_SharedClassName(key := some_key, attributes...);
```

External CIs
Similarly, it is sometimes necessary to specify a relationship to a CI that is not part of the target subgraph. An example is to relate the `BMC_ComputerSystem` for a physical host to the one for a virtual host — the two CIs belong to different subgraphs. External CIs are specified with a function in the `sync.external` namespace:

```plaintext
value := sync.external.BMC_ClassName(key := some_key [, namespace := "_NAME_"
```

The key must be specified, and namespace must be specified if required. No other attributes can be set.

It is not an error if a CI with the specified key does not exist in the CMDB. In that situation, the CI and any relationships to it are simply ignored.

Cross-reference CIs
As the mapping is processed, the CIs are specified in a tree traversal across the graph. To refer to a CI specified in a different branch of the tree, it can be specified with a function in the `sync.crossref` namespace:

```plaintext
value := sync.crossref.BMC_ClassName(key := some_key [, namespace := "_NAME_"
```

The key must be specified, and namespace must be specified if required. No other attributes can be set.

It is a runtime error to specify a cross-reference to a CI that is not fully specified elsewhere within the mapping.
Relationship definition

Relationships are specified with functions in the sync.rel namespace:

```
sync.rel.BMC_RelName(Source := source_val,
                       Destination := dest_val,
                       Name := "RELNAME" [, ... ]);
```

The first two parameters must be Source and Destination. Any other attributes can also be set; Name is not required, but it is conventionally always set.

The Impacted and ImpactWeight attributes can be used to create impact relationships. They specify impact direction and percentage respectively. For example, an ImpactWeight of 25 might be appropriate when representing an impact relationship between a BMC_ComputerSystem and BMC_Cluster, where there are four computers in a cluster. The value assigned to Impacted must be the Source or Destination CI appearing in the same definition.

```
sync.rel.BMC_Component(
    Source       := computersystem,
    Destination  := cluster_ci,
    Name         := "CLUSTEREDSYSTEM",
    Impacted     := cluster_ci,
    ImpactWeight := cluster_rel.impact_weight
);
```

Note that prior to tpl 1.10 multiple relationship definitions were needed to represent a single impact relationship in the CMDB. This was because the way in which impact is represented in the CMDB depends on the data model in effect (see below). In.tpl 1.10 and later, only one definition is required.

Traversal looping

One of the main activities performed in the body is to iterate over the nodes reached through the traversals specified in the mapping block. A for each loop is used to iterate over the named nodes:

```
for each source_node do ...end for
```

The nesting structure of for each loops in the body must match the nesting structure of the traverse expressions in the mapping block.

A for each loop is required even if the corresponding traversal is expected to reach just one node. There is no other way to access the state of the traversed-to node.
**Group block**

When the mapping block specifies a group, there must be a corresponding group block in the body. The group block will always be inside a for each block, either directly within a single syncmapping body or in an extended source syncmapping.

A group block takes the form:

```plaintext
for each traversed_to_node do ...
    ident := group_identifier; group group_name with ident do ...
    for each expand_name do...
    ...end for; end group; end for;
```

The grouping is evaluated in two phases. In the first phase, every iteration of the surrounding for each loop is executed. The nodes are grouped according to the identifier provided to the group expression. After all the iterations of the for each loop, the group block is executed once for each group. If the group declaration in the mapping block contains an expand declaration, there should be a corresponding for each loop in the body.

The group content is executed in a context based on an arbitrary member of the group. Any local variables from the surrounding loop will therefore be valid for a member of the group, but there is no guarantee that it will be the same group member each time a particular group is processed.

**Data models**

Different versions of the CMDB have subtly different data models. Syncmappings can support multiple data models with datamodel declarations. CMDB data models are assigned simple integer values:

<table>
<thead>
<tr>
<th>Data model</th>
<th>CMDB versions</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>7.6.03 and later</td>
<td>HasImpact and ImpactDirection attributes are set as appropriate.</td>
</tr>
<tr>
<td>5</td>
<td>7.6.03 and later</td>
<td>Only to be used with legacy SIM version 7.4. BMC_Impact relationships with Name “ImpactOnly” are created.</td>
</tr>
<tr>
<td>4</td>
<td>7.6.03 and later</td>
<td>No impact details are set by BMC Atrium Discovery. They may be set by Impact Normalization in the CMDB.</td>
</tr>
<tr>
<td>3</td>
<td>7.6 before 7.6.03</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
<tr>
<td>2</td>
<td>7.5</td>
<td>BMC_Impact relationships with name “IMPACT” are created.</td>
</tr>
</tbody>
</table>

A syncmapping can limit itself to a particular set of data models with a datamodel declaration in the overview:

```plaintext
overview tags Some_tags; datamodel 3, 4; // Only CMDB 7.6.x
end overview
```
The body of a syncmapping can further modify its behavior for different data models with a datamodel block. The datamodel block only executes if the data model in effect matches the declaration:

```plaintext
ci := sync.BMC_Thing(key := my_key, ...);
datamodel 2, 3 do  
  sync.rel.BMC_Dependency(   
    Source := ci, 
    Destination := other_ci, 
    Name := "DEPENDENCY_NAME"  
  );
end datamodel;
```

The data model in effect is not chosen automatically. After you configure CMDB synchronization, the data model is selected. This is described in Setting up a CMDB synchronization connection.

### TPL Keywords

The following are TPL keywords:

- and
- as
- body
- break
- by
- configuration
- constants
- continue
- default
- define
- defined
- definitions
- desc
- do
- elif
- else
- end
- enumeration
- exists
- expand
- explode
- false
- flags
- for
- from
- has
• identify
• if
• import
• in
• is
• locale
• matches
• metadata
• module
• nodes
• nodecount
• none
• not
• on
• or
• order
• out
• overrides
• overview
• pattern
• processwith
• relationship
• removal
• requires
• search
• show
• step
• stop
• substring
• subword
• summary
• table
• tags
• taxonomy
• then
• tpl
• traverse
• triggers
• true
• where
• with
• void

**TPL Grammar**

**Basic lexical rules**

```plaintext
<id> ::= \$?[a-zA-Z_]\[a-zA-Z_0-9]*
<ids> ::= <id> ('.' <id>)*
<number> ::= 0x[0-9a-fA-F]+
               | [0-9]+
<string> ::= { "" anything except newline or unescaped double quote ""
                | """" anything other than three double quotes """
                | \ anything except newline or unescaped single quote \n                | \"\"\" anything other than three single quotes \"\"\"
<qualified_string> ::= <string>
                    | <id> <base_string>
<boolean> ::= true
              | false
<uri> ::= [a-zA-Z]+':[a-zA-Z0-9_.-%%3aNo.]+
<traversal> ::= <id>? ':' <id>?
               | <traversal> ':' <id>?
```

**Aliases for readability**

```plaintext
<description> ::= <qualified_string>
<module_name> ::= <ids>
<tag_name> ::= <ids>
<pattern_name> ::= <id>
<table_name> ::= <id>
<identify_name> ::= <id>
<node_kind> ::= <id>
<attribute_name> ::= <id>
<variable> ::= <ids>
<scope> ::= <ids>
<function> ::= <ids>
<version_number> ::= <number> '.' <number>
```

**Module structure**

```plaintext
<module> ::= tpl <version_number> module <ids> ';'
            <metadata>?
            ( <definition> ';' )+
<metadata> ::= metadata
            ( <id> ::= <qualified_string> )+
            end metadata;
```
<definition> ::= <import>
  | <pattern>
  | <table>
  | <identify>

<import> ::= from <module_name> import <id> <version_number> (as <id>)?
  (',' <id> <version_number> (as <id>)?)*

Tags

<tags> ::= tags <tag_name> (',' <tag_name>)*

Top-level declarations

<pattern> ::= pattern <pattern_name> <version_number>
  <description>
  <metadata>?
  <overview>
  <constants>?
  <triggers>
  <pattern_body>
  <removal>*
  end pattern
<table> ::= table <table_name> <version_number>
  <table_row>*
  end table
<identify> ::= identify <identify_name> <version_number>
  <tags> ';'
  <node_kind> <attribute_name> (',' <attribute_name>)+ '->'
  <attribute_name> (',' <attribute_name>)+ ';'
  <table_row>*
  end identify
<table_row> ::= <table_key> '->' <table_value> ';'
  | default '->' <table_value> ';'
<table_key> ::= <table_value>
<table_value> ::= <constant_expression> (',' <constant_expression>)*

Overview

<overview> ::= overview
  ( <overview_entry> ';')*
  end overview
<overview_entry> ::= <implements>
  | <overrides>
<overrides> ::= overrides <pattern_name> ( ',' <pattern_name> )*
<requires> ::= requires <pattern_name> ( ',' <pattern_name> )*

Constants

<constants> ::= constants
             ( <pattern_constant> )+
             end constants ';
<pattern_constant> ::= <id> ':=' <constant_expression> ';

Triggers

<triggers> ::= triggers (scope <node_kind>)?
           <trigger_condition>*
             end triggers ';
<trigger_condition> ::= on <trigger_match_list> ';
<trigger_match_list> ::= <trigger_match_kind>
<trigger_match_kind> ::= <id> ':=' ( relationship )? <id> <trigger_fires_list>?
                       where <trigger_match>
<trigger_fires_list> ::= <trigger_fires> ( ',' <trigger_fires> )*
<trigger_fires> ::= created
                  | modified
                  | destroyed
                  | confirmed
<trigger_match> ::= <trigger_match_and>
<trigger_match_and> ::= <trigger_match_attribute>
                      | <trigger_match_and> and <trigger_match_attribute>
<trigger_match_attribute> ::= <id> '=' <constant_expression>
                          | <id> exists
                          | <id> in <constant_expression>
                          | <id> matches <constant_expression>
                          | '(' <trigger_match_and> ')'

Body

<pattern_body> ::= body
                 <statements>
                 end body ';

Removal

\[
<\text{removal}> ::= \text{removal} \\
\quad \text{on} \ <\text{variable}> \ '=>' \ <\text{node}\_\text{kind}> \ <\text{removal}\_\text{kind}> \ ( \text{where} \ <\text{trigger}\_\text{match}> \ )? ';' \\
\quad <\text{statement}\_\text{list}> \\
\quad \text{end} \ \text{removal} ';' \\
<\text{removal}\_\text{kind}> ::= \text{aged} \\
\quad \text{unconfirmed}
\]

Statements

\[
<\text{statement}\_\text{list}> ::= \{ <\text{statement}> ';'> \} \\
<\text{statement}> ::= <\text{empty}\_\text{statement}> \\
\quad | \ <\text{assignment}\_\text{statement}> \\
\quad | \ <\text{function}\_\text{call}> \\
\quad | \ <\text{if}\_\text{statement}> \\
\quad | \ <\text{for}\_\text{statement}> \\
\quad | \ <\text{break}\_\text{statement}> \\
\quad | \ <\text{continue}\_\text{statement}> \\
\quad | \ <\text{stop}\_\text{statement}>
\]

Assignment

\[
<\text{assignment}\_\text{statement}> ::= <\text{lvalue}\_\text{list}> \ '=>' \ <\text{logical}\_\text{expression}> \\
<\text{lvalue}\_\text{list}> ::= <\text{lvalue}> \ ( ',' <\text{lvalue}> )* \\
<\text{lvalue}> ::= <\text{ids}> \\
\quad | \ <\text{lvalue}> '[' <\text{logical}\_\text{expression}> ']' \\
\quad | \ <\text{lvalue}> '\[' <\text{logical}\_\text{expression}> '\]
\]

If

\[
<\text{if}\_\text{statement}> ::= \text{if} \ <\text{logical}\_\text{expression}> \ \text{then} \ <\text{statements}> \\
\quad (\text{elif} \ <\text{logical}\_\text{expression}> \ \text{then} \ <\text{statements}>)* \\
\quad (\text{else} \ <\text{statements}>)? \\
\quad \text{end} \ \text{if}
\]
For

```markdown
<for_statement> ::= for <variable> in <logical_expression> do
    <statements>
end for
<break_statement> ::= break
<continue_statement> ::= continue
```

Stop

```markdown
<stop_statement> ::= stop
```

Functions

```markdown
<function_call> ::= <function> '(' <arg_list>? ')' 
<arg_list> ::= <arg_spec> (',' <arg_spec>)* 
<arg_spec> ::= <logical_expression> 
    | <id> ':=' <logical_expression>
```

Expand expression

```markdown
<expand_expression> ::= expand '(' <arg_list> ')
```

Expressions

```markdown
<logical_expression> ::= <and_test>
    | <logical_expression> or <and_test>
<and_test> ::= <not_test>
    | <and_test> and <not_test>
<not_test> ::= <comparison>
    | not <not_test>
<comparison> ::= <or_expression>
    | <or_expression> <relational> <or_expression>
    | '*' <relational> <or_expression>
    | <or_expression> is? not? defined
<relational> ::= '<' | '<=' | '>' | '>=' | '=' | '<>' | in
    | not in | has subword | has substring | matches
```
Documentation

BMC Software Confidential

<or_expression>

::= <xor_expression>

<xor_expression>

| <or_expression> '|' <xor_expression>
::= <and_expression>
| <xor_expression> '^' <and_expression>

<and_expression>

::= <shift_expression>

<shift_expression>

| <and_expression> '&' <shift_expression>
::= <add_expression>
| <shift_expression> '<<' <add_expression>
| <shift_expression> '>>' <add_expression>

<add_expression>

::= <multiply_expression>
| <add_expression> '+' <multiply_expression>
| <add_expression> '-' <multiply_expression>

<multiply_expression> ::= <unary_expression>
| <multiply_expression> '*' <unary_expression>
| <multiply_expression> '/' <unary_expression>
| <multiply_expression> '%' <unary_expression>
<unary_expression>

::= <primary>
| '-' <unary_expression>
| '+' <unary_expression>
| '~' <unary_expression>

<constant_expression> ::= <logical_expression>
<primary>

::= <atom>
| <function_call>
| <search_expression>
| <expand_expression>
| <subscript>
| <list_expression>
| <attribute>
| <keyword_expression>

<subscript>
<list_expression>

| <group_expression>
::= <primary> '[' <logical_expression> ']'
::= '[' ( <logical_expression> ',' )* <logical_expression>? ']'

<scoped_attribute>

::= <scope> '.' <attribute>

<attribute>

::= <ids>
| <ids> '.' <key_expression>
<key_expression> ::= '#'
| '##'
| '#id'
| '#' <traversal>
| '#' <traversal> '.' <attribute>
| '#' <string> '(' <attributes> ')'
<keyword_expression>
<group_expression>
<atom>

::= structure
::= '(' <expression> ( ',' <expression> )* ','? ')'
::= none
| <bool>
| <qualified_string>
| <number>

BMC Atrium Discovery 10.2

Page 1615 of 1671


Search expression

<search_expression> ::= search '('
    <flags>?  
    <search_in>? 
    <kind_list> 
    <with_funcs>? 
    <where_clause>? 
    <traversals>? 
    <order_by_clause>? 
    <locale>? 
    <show>? 
    <search_process_with>?
    ')'

<flags> ::= flags '{' <id> ( ',' <id> )* '}'
<search_in> ::= in <id>
                   | in <string>
<kind_list> ::= '*'
                   | <kind_comma_list>
<kind_comma_list> ::= <node_kind> ( ',' <node_kind> )*

with functions

<with_funcs> ::= with <with_func> ( ',' <with_func> )* 
<with_func> ::= <id> '(' <search_expression_commalist> ')' as <id>

traversal

<traversals> ::= <traverse_or_expand>+
<traverse_or_expand> ::= <traverse_clause>
                      | <expand_clause>
                      | <step_in_clause>
                      | <step_out_clause>
<traverse_clause> ::= traverse <traversal> <with_funcs>? <where_clause>? 
<expand_clause> ::= expand <traversal> <with_funcs>? <where_clause>? 
<step_in_clause> ::= step in <traversal> <with_funcs>? <where_clause>? 
<step_out_clause> ::= step out <traversal> <with_funcs>? <where_clause>?

where

<where_clause> ::= where <search_logical_expression>
order by

<order_by_clause> ::= order by <order_by_commalist>
   | order by taxonomy
<order_by_commalist> ::= <order_by> ( ',' <order_by> )*  
<order_by> ::= <search_value_expression> <direction>?
<direction> ::= desc

locale

<locale> ::= locale <string>
   | locale <id>

show

<show> ::= show <show_list>?
<show_list> ::= <show_clause> ( ',' <show_clause> )*  
<show_clause> ::= <search_value_expression> <as>?
   | explode <search_value_expression> <as>?
   | summary
   | '*'
<as> ::= as <id>
   | as <string>
<search_logical_expression> ::= <search_and_expression>
   | <search_logical_expression> or <search_and_expression>
<search_and_expression> ::= <search_and_expression>
   | <search_and_expression> and <search_not_expression>
<search_not_expression> ::= <search_comparison>
   | not <search_not_expression>
<search_comparison> ::= <search_value_expression> '=' <search_value_expression>
   | <search_value_expression> '>' <search_value_expression>
   | <search_value_expression> '>=' <search_value_expression>
   | <search_value_expression> '<' <search_value_expression>
   | <search_value_expression> '<=' <search_value_expression>
   | <search_value_expression> has subword <search_value_expression>
   | <search_value_expression> has substring <search_value_expression>
   | <search_value_expression> matches <search_value_expression>
   | <search_value_expression> not matches <search_value_expression>
   | <search_value_expression> in <search_value_expression>
   | <search_value_expression> not in <search_value_expression>
   | <search_value_expression> defined
   | <search_value_expression> is defined
| <search_value_expression> not defined
| <search_value_expression> is not defined
| '* '=' <search_value_expression>
| '* has subword <search_value_expression>
| '* has substring <search_value_expression>
| '* matches <search_value_expression>
| <search_value_expression>

<search_value_expression> ::= <search_multiply_expression>
| <search_value_expression> '+' <search_multiply_expression>
| <search_value_expression> '-' <search_multiply_expression>

<search_multiply_expression>
::= <search_unary_expression>
| <search_multiply_expression> '/' <search_unary_expression>
| <search_multiply_expression> '*' <search_unary_expression>

<search_unary_expression> ::= <search_primary_expression>
| '-' <search_unary_expression>

<search_primary_expression> ::= <search_value>
| <id> '(' <search_expression_commalist>? ')'
| '[' <search_expression_commalist>? ']
| '(' <search_logical_expression> ')'

<search_value> ::= nodecount '(' <traversals>? ')'
| nodes '(' <traversals>? ')
| <number>
| <string>
| <search_attribute>

<search_attribute_commalist> ::= <search_attribute> ( ',' <search_attribute> )*

<search_attribute> ::= <id>
| <search_key_expression>
| <search_at_expression>

<search_key_expression> ::= '#'
| '##'
| '#id'
| '# <traversal>
| '# <traversal> '.' <search_attribute>
| '# <string> '(' <search_attribute_commalist>? ')''

<search_expression_commalist> ::= <search_value_expression> ( ',', <search_value_expression> )*

<search_process_with> ::= processwith <search_process_with_func_list>
<search_process_with_func_list> ::= <search_process_with_func>
| <search_process_with_func_list> ',' <search_process_with_func>
<search_process_with_func> ::= <id>
| <id> '(' <search_expression_commalist>? ')'[

### String Escape Characters

Unqualified string literals support the following escape sequences:

<table>
<thead>
<tr>
<th>Escape sequence</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| \
<p>| \ \ |</p>
<table>
<thead>
<tr>
<th>Escape sequence</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>'</td>
<td>Single quote ('')</td>
</tr>
<tr>
<td>&quot;</td>
<td>Double quote (&quot;)</td>
</tr>
<tr>
<td>\a</td>
<td>ASCII bell (BEL)</td>
</tr>
<tr>
<td>\b</td>
<td>ASCII Backspace (BS)</td>
</tr>
<tr>
<td>\f</td>
<td>ASCII Formfeed (FF)</td>
</tr>
<tr>
<td>\n</td>
<td>ASCII Linefeed (LF)</td>
</tr>
<tr>
<td>\r</td>
<td>ASCII Carriage Return (CR)</td>
</tr>
<tr>
<td>\t</td>
<td>ASCII Horizontal Tab (TAB)</td>
</tr>
<tr>
<td>\uxxxxx</td>
<td>16-bit hex Unicode character (encoded in Unicode or UTF8 depending upon string type)</td>
</tr>
<tr>
<td>\Uxxxxxxxxxxx</td>
<td>32-bit hex Unicode character (encoded in Unicode or UTF8 depending upon string type)</td>
</tr>
<tr>
<td>\v</td>
<td>ASCII Vertical Tab (VT)</td>
</tr>
<tr>
<td>\ooo</td>
<td>Character with octal value \ooo (1-3 octal digits)</td>
</tr>
<tr>
<td>\xhh</td>
<td>Character with hex value \xhh (1-2 hex digits)</td>
</tr>
</tbody>
</table>

Using TPL to enrich discovered data

While standard discovery processes can provide most key information, there are times when you might want to track information that is not available to discovery. A good example of this type of information is the exact location of a host. In most situations, there is nothing discoverable on a host that allows you to know that it is located in U23 of Rack 15 in Data Center 2 of the Houston campus. While the most common example of this is to track the location of hosts, similar approaches could be employed for relating hosts to organizational units, or owners.

BMC Atrium Discovery makes it very easy for you to extract information from external data sources and include it in its own data. Commonly, customers can use these capabilities to:

- Relate hosts to locations using a host naming scheme
- Reading location information from a file or registry setting
- Using a table in a pattern to relate subnets to locations
- Querying an SQL database for location information

These are described in the sections below.

In the TPL code (patterns) snippets, logging and basic error checking is shown in line with BMC Software recommendations. When you are developing patterns, you need to run them outside a normal discovery run against previously scanned host. For information on doing this, see Executing patterns manually.
Location using a naming scheme

The `template_host_location` template pattern shows you how to relate hosts to locations based on hostname. This pattern relies on a naming scheme in which hostnames contain a location identifier such as `lon` for London, or `hou` for Houston.

See Pattern templates for a description of all template patterns supplied on the appliance.

Reading location from a file

A common way of incorporating location information into a host is to store a file in the host's file system. The `template_si_version_xml_file` template pattern demonstrates taking information from an XML file to version an SI. The same approach can be used to extract location information from an XML file.

If the information is stored in an XML file use xpath as shown in the template pattern. The TPL xpath functions are described in the TPL Guide. In the case where information is placed in a flat file, a simple regex is the most effective method of extracting the location. The TPL regex functions are described in the TPL Guide.

The full specification of the regular expression syntax used in BMC Atrium Discovery can be found on the Python website. A simpler introduction is also available.

Here is an example XML snippet for a file:

```xml
<?xml version="1.0"?>
<config>
  <physical>
    <location="Houston">
      ...
    </location>
  </physical>
</config>
```

Here is the TPL which retrieves the file and uses xpath to obtain the location from the file:

```plaintext
// Get config file and check that it was retrieved successfully
cfgfile := discovery.fileGet(host, cfgfilePath);
if cfgfile then
  // Extract location from conf file
  location := xpath.evaluate(cfgfile.content, raw '/config/physical/@location');
```

Now you can create or update the relationship between the host and location node. The following code snippet uses the `model.uniquerel` function to create or update the relationship.
// Relate host to location. Using "uniquerel" rather than "rel" means that
// any existing Location relationships between this Host (the first
// parameter) and any Location nodes other than the one given (the second
// parameter) are removed. If a host has changed location, this keeps the
// model up-to-date.
log.info("Subnet_2_Locations: model.uniquerel.Location %host% %location%");
model.uniquerel.Location(
    ElementInLocation := host,
    Location := location
);

See the template_si_version_xml_file template pattern for an example of how these
statements are used in the pattern.

Reading location from a registry key

Some administrators might prefer to place information in a registry setting rather than the files used
in the previous section. The template pattern template_si_version_registry demonstrates
extracting information from the registry to version an SI. The same approach can be used to
extract location information from the registry.

// Get the registry key
location := discovery.registryKey
    (process, "HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\Location")

Using a table in a pattern to relate subnets to locations

Subnets can be used to identify locations of hosts. You can extend the
template_host_location pattern to map subnets to locations. The following TPL snippet
shows how you could use a table to map the subnet to hosts. You could also hard code a mapping
of hostnames to locations if that information is available. See TPL Guide for more information on
tables.

table SubnetLocations 1.0
    "101.10.10.0/24" -> "London Victoria";
    "101.10.11.0/24" -> "London Egham";
    "137.72.94.0/24" -> "Houston";
    "137.72.95.0/24" -> "Dallas";
    default -> "unknown";
end table;

The following TPL snippet shows the search from the host to associated subnet or subnets.
### Querying an SQL database for location information

The `template_sql_asset_integration` template pattern is supplied with BMC Atrium Discovery which enables you to extract location information from an SQL database. To query a database you must define the asset database in the pattern and activate the pattern. After activation, this creates an integration point into which you add connection information. For information on using integration points to connect to databases, see Integration points.

### Automatically creating locations

The pattern examples above have shown a number of ways of determining locations and how to create the relationship between host and existing location nodes. The following code snippet shows how you can create or update a location node, asserting that the location exists.

```plaintext
// Create or update a Location node for this. For a list of attributes for // Location nodes, see Administration > Taxonomy. Every attribute used on // the node in the "location_row" variable must be returned by the query's // SELECT statement.
log.debug('Setting location of %host.hostname% as %location_row.address%');

location_node := model.Location(  
  key := text.hash("%location_row.name% %location_row.address%"),  
  name := location_row.name,
```
address := location_row.address
);  

VMware ESX and ESXi hosts discovery using custom patterns

You can develop your custom patterns to discover information about the VMware ESX and ESXi hosts from the VMware vSphere API.

This page contains information about the following:

- **Conditions** required for successful discovery of the VMware ESX and ESXi hosts.
- **Queries** used by the patterns to retrieve information about the VMware ESX and ESXi hosts.

To learn how to upload patterns to the appliance, activate or deactivate patterns on the appliance, and delete patterns which are no longer required, see Knowledge management.

**Conditions for successful VMware ESX and ESXi hosts discovery**

For successful VMware ESX and ESXi hosts discovery, you must ensure the following conditions:

- Define the VMware vCenter and vSphere discovery credentials. For more information, see Configuring vCenter credentials and Managing vSphere credentials.

⚠️ BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery) uses the vSphere API to communicate directly with VMware ESX and ESXi hosts where no VMware vCenter credentials are available. Running a large number of VMware vSphere or vCenter queries can have performance impact on both the VMware vCenter server and the corresponding VMware vSphere hosts. To avoid this, the API must be used sparingly.

- While developing your custom patterns, ensure the following:
  - For VMware vCenter, in the definition block, type must be set to `vcenter_discovery` and group must be set to "vCenter".
    ```java
type := vcenter_discovery;
group := "vCenter";
```
  - For VMware vSphere, in the definition block, type must be set to `vsphere_discovery` and group must be set to "vSphere".
    ```java
type := vsphere_discovery;
group := "vSphere";
```
type := vsphere_discovery;
group := "vSphere";

- By default, BMC Atrium Discovery retrieves the vsphere_id and vsphere_type attributes for the requested ManagedObjectReference (MOR). To refer to an MOR property in your pattern, you must replace the periods (.) with underscores (_). For example, in the pattern, you can refer to the host_system.hardware.systemInfo.uuid attribute as host_system.hardware_systemInfo_uuid.

⚠️ Currently, complex MOR attributes are not retrieved by BMC Atrium Discovery.

- BMC Atrium Discovery records the vCenter query error and query result in tw_svc_vsphere.log. For example, if you specify HostStorageSystem in the findObject query, the error in vmware sdk will be recorded in tw_svc_vsphere.log:

```
Provider.threadPool1: 2013-08-29 20:31:26,510:
com.tideway.integrations.vsphere.ProviderSession: INFO: Executing DIP query:
queryState:{object_type=HostStorageSystem, parameters=[],
properties=[storageDeviceInfo.multipathInfo.lun.id],
vcenter_only=true, query=[findObjects]}
queryParams:{}
state:{port=443, username=vmreadonly, vcenter_ip=137.72.94.9,
credential_id=dc70aa30f2f269288532ac16220b0b11,
endpoint=137.72.94.9, timeout=60000}
Provider.threadPool1: 2013-08-29 20:31:26,662:
com.tideway.integrations.vsphere.Connection: ERROR: 137.72.94.9:
Failed to get View: com.vmware.vim25.InvalidArgument
```

**VMware ESX and ESXi hosts discovery queries**

In BMC Atrium Discovery versions earlier than 9.0, the following queries are used by the patterns for VMware ESX and ESXi hosts discovery:

- findObjects
- traverseObjects

In BMC Atrium Discovery 9.0 and later versions, in addition to the above queries, the following queries have been newly added:

- getPropertyTable
- getProperties
findObjects

It queries to search from the root folder the instances of an object type and returns the requested properties for each object found. The required parameters are:

- `query`: findObjects
- `object_type`: The object type must be a valid container view specified in the VMware documentation. For example, Folder, Datacenter, ComputeResource, ResourcePool, or HostSystem.
- `properties`: The properties to retrieve must be simple type (string, int, boolean, and so on), no object, or list.

Following is an example of using findObjects to get name and hardware.systemInfo.uuid of HostSystem MOR:

```html
// You must define the function
definitions vCenter 1.0
    """DIP queries for getting HostSystem information"

    type := vcenter_discovery;
    group := "vCenter";

    define getHosts
        'return all the hosts managed by vCenter'
        query := "findObjects";
        object_type := "HostSystem";
        properties := "name", "hardware.systemInfo.uuid";
        vcenter_only := true;
    end define;
end definitions;

// In the pattern body, you use the function to retrieve name and
// hardware.systemInfo.uuid properties of HostSystem MOR
hosts := vCenter.getHosts(endpoint:=vc_host);
if hosts then
    for host_info in hosts do
        host_vsphereid := host_info.vsphere_id;
        host_name := host_info.name;
        host_uuid := host_info.hardware.systemInfo.uuid;
        log.debug("found Host name=%host_name%,
                   vsphere_id=%host_vsphereid%,
                   uuid=%host_uuid%");
    end for;
end if;
```

⚠️ vcenter_only flag

// BMC Software Confidential
Page 1625 of 1671
The vcenter_only flag in the above example means that the call is intended to talk only to VMware vCenter instances and will fail if directed at a VMware ESXi host. This is required because both the ESXi hosts and vCenter use the same API, but some calls only make sense when talking to vCenter (for example, listing all the managed hosts). For most queries, it must be false. Otherwise, the calls will fail if an ESX host is called directly.

You can see the result of the findObjects query in the tw_vsphere_log. Following is the result of the findObjects query from the above example:

```
com.tideway.integrations.vsphere.Connection:
INFO: Create authenticated connection to https://137.72.94.9/sdk/vimService
Provider.threadPool1: 2013-09-27 21:13:56,790:
com.tideway.integrations.vsphere.ProviderSession:
INFO: Executing DIP query:
queryState:{object_type=HostSystem, parameters=[],
properties=[name, hardware.systemInfo.uuid],
vcenter_only=true, query=[findObjects]}
queryParams:{}
state:{port=443, username=vmreadonly, vcenter_ip=137.72.94.9,
credential_id=dc70aa30f2f269288532ac162220b0b11, endpoint=137.72.94.9,
timeout=60000}
Provider.threadPool1: 2013-09-27 21:13:58,817:
com.tideway.integrations.vsphere.ProviderSession:
INFO: returning ([-- 1: vSphereIntegrationResult
  hardware_systemInfo_uuid : 44454c4c-5900-104c-8051-c7c04f333253
    name : pun-esx-rds36.bmc.com
    vsphere_id : host-1867
    vsphere_type : HostSystem
.....
```

**traverseObjects**

It queries to traverse from the initial object to instances of an object type and get properties on those objects. You will need to know the vsphere_id (MOR ID) from Managed Object Type that you want to start traversing with. It returns the requested properties for each object found. The required parameters are:

- query: traverseObjects
- from_type: The initial MOR object type.
- object_type: The Object type to traverse to.
- properties: All the properties to be retrieved must be of simple type.
- traversal: The traversal path
- parameters: The vsphere_id of the initial MOR.

Following is an example of retrieving Datastore objects from HostSystem objects:
definitions vCenterDef 2.0

  "DIP queries for getting storage information"

  type := vcenter_discovery;
group := "vCenter";

define getHosts
  'return all the hosts managed by vCenter'
  query := "findObjects";
  object_type := "HostSystem";
  properties := "name";
  vcenter_only := true;
end define;

define getStorageInHost
  'return the datastore in a given host. Pass in vsphere_id of the host as from_id'
  query := "traverseObjects";
  from_type := "HostSystem";
  object_type := "Datastore";
  traversal := "datastore";
  properties := "name";
  parameters := from_id;
  vcenter_only := true;
end define;
end definitions;

pattern vCenterTest 1.0

  triggers
    on si := SoftwareInstance created, confirmed where
      type = 'VMware VirtualCenter Server' or
      type = 'VMware vCenter Server';
  end triggers;

  body

  host := model.host(si);
  log.debug("VCTEST Enter getHostStorageSystem");
  hosts := vCenterDef.getHosts(endpoint:=host);
  if hosts then
    for host_info in hosts do
      host_id := host_info.vsphere_id;
      host_name := host_info.name;
      log.debug("VCTEST get Host_ID %host_id% name %host_name");
      storages := vCenterDef.getStorageInHost(endpoint:=host, from_id:=host_id);
      if storages then
        for storage_info in storages do
          storage_id := storage_info.vsphere_id;
          storage_name := storage_info.name;
          log.debug("VCTEST get datastore id %storage_id% datastore_name %stor
getPropertyTable

It queries to retrieve a table of values from a given MOR and is intended to be used to retrieve nested properties from lists and arrays. The query returns multiple NodeState objects, one for each DataObject in the list or array. The required parameters are:

- query: "getPropertyTable"
- parameters: mor, type, property, and columns
  - mor: The MOR id (vsphere_id)
  - type: The MOR object type
  - property: The parent property path of MOR column
  - columns: The property of MOR (It must not be the composite column name. For example, key or deviceName is a valid column. However, capacity.block is not a valid column.)

Following is an example of retrieving multipathing or LUN data properties from a storage device:

```plaintext
definitions vCenterDef 1.0
  """DIP queries for getting LUN information""

  type := vcenter_discovery;
  group := "vCenter";

  define getHosts
    'return all the hosts managed by vCenter'
    query := "findObjects";
    object_type := "HostSystem";
    properties := "name";
    vcenter_only := true;
  end define;

  define getDiskInfo
    'return disk information'
    query := "getPropertyTable";
    parameters := mor, type, property, columns;
  end define;

end definitions;

// In the body of the pattern, you define
// the parameters to be retrieved by this query
// First, Let's get the HostSystems managed by the vCenter.
hosts := vCenterDef.getHosts(endpoint:=host);
if hosts then
  for host_info in hosts do
    host_id := host_info.vsphere_id;
    host_name := host_info.name;

    // Example to get scsiLun (serial number and device name) for HostSystem
    disks := vCenterDef.getDiskInfo(endpoint := host,
      mor := host_id,
      type := "HostSystem",
      property := "config.storageDevice.scsiLun",
      columns := ["serialNumber", "deviceName"]
    );
    log.debug("VCTEST get scsiLun for %host_id% name %host_name% ");
    if disks then
      for disk_info in disks do
        log.debug("VCTEST get scsiLun serialnumber %disk_info.serialNumber% ");
        log.debug("VCTEST get scsiLun deviceName %disk_info.deviceName% ");
      end for;
    end if;

    // Example to get multipathing / LUN data properties from a storage device
    disks := vCenterDef.getDiskInfo(endpoint := host,
      mor := host_id,
      type := "HostSystem",
      property := "config.storageDevice.multipathInfo.lun",
      columns := ["key"]
    );
    log.debug("VCTEST get multipathInfo.lun %host_id% name %host_name% ");
    if disks then
      for disk_info in disks do
        log.debug("VCTEST get multipathInfo.lun key %disk_info.key% ");
        // For each lun, get the path name
        luns := vCenterDef.getDiskInfo(endpoint := host,
          mor := host_id,
          type := "HostSystem",
          property := "config.storageDevice.multipathInfo.lun["%disk_info.key%","name"]",
          columns := ["name"]
        );
        log.debug("VCTEST get multipathInfo.lun.path %host_id% name %host_name% ");
        if luns then
          for lun_info in luns do
            log.debug("VCTEST get multipathInfo.lun.path %lun_info.name% ");
          end for;
        end if;
      end for;
    end if;
  end for;
end if;
getProperties

It queries to retrieve properties from a given MOR and returns the requested properties for each object found. The required parameters are:

- `query`: = "getProperties";
- `parameters`: = mor, type, attributes;

Following is an example of retrieving scsiLun data properties from a storage device:

⚠️ As scsiLun is an array, you must retrieve the scsiLun key first and then the attribute for each key. If the attribute does not exist, then BMC Atrium Discovery may throw an ECA Error in `tw_svc_reasoning.log`.

```plaintext
.....
define getPropertyInfo
  'retrieve properties from a given MOR'
  query := "getProperties";
  parameters := mor, type, attributes;
end define;

define getPropTable
  'retrieve a table of values from a given MOR properties'
  query := "getPropertyTable";
  parameters := mor, type, property, columns;
end define;
.....
hosts := vCenterDef.getHosts(endpoint:=host);
if hosts then
  for host_info in hosts do
    host_id := host_info.vsphere_id;
    host_name := host_info.name;
    disks := vCenterDef.getPropTable(endpoint := host,
      mor := host_id,
      type := "HostSystem",
      property := "config.storageDevice.scsiLun",
      columns := ["key"]);
    log.debug("VCTEST get scsiLun %host_id% name %host_name%")
    if disks then
      for disk_info in disks do
        log.debug("VCTEST get scsiLun key %disk_info.key%")
        scsiLuns := vCenterDef.getPropertyInfo(endpoint := host,
          mor := host_id,
          type := "HostSystem",
          attributes := ["config.storageDevice.scsiLun["%disk_info.key%"].dev
        );
      
    
```
Documentation

BMC Software Confidential

if scsiLuns then
for scsiLun in scsiLuns do
vsphere_id := scsiLun["vsphere_id"];
log.debug("VCTEST get scsiLun vsphereid %vsphere_id%");
keyDeviceName := "config_storageDevice_scsiLun[\"%disk_info.key%\"]_deviceName";
attrName := text.replace(keyDeviceName, '.', '_');
devName := scsiLun["%attrName%"];
log.debug("VCTEST get scsiLun devName %devName%");

keyCapName := "config_storageDevice_scsiLun[\"%disk_info.key%\"]_capabilities_updat
attrName := text.replace(keyCapName, '.', '_');
capName := scsiLun["%attrName%"];
log.debug("VCTEST get scsiLun capName %capName%");
end for;
end if;
end for;
end if;
end for;
end if;

You can see the result of the getProperties query in the tw_vsphere_log. Following is an example
of the result of the getProperties query in the example:

Provider.threadPool1: 2013-09-27 21:14:01,736:
com.tideway.integrations.vsphere.ProviderSession:
INFO: Executing DIP query:

queryState:{query=[getProperties], parameters=[mor, type, attributes], properties=[]}
queryParams:{type=HostSystem, mor=host-1867,
attributes=[config.storageDevice.scsiLun["key-vim.host.ScsiLun-0005000000766d686261333a303a30"]

config.storageDevice.scsiLun["key-vim.host.ScsiLun-0005000000766d686261333a303a30"].capabilities.updateDispl
state:{port=443, username=vmreadonly, vcenter_ip=137.72.94.9,
credential_id=dc70aa30f2f269288532ac16220b0b11, endpoint=137.72.94.9, timeout=60000}
Provider.threadPool1: 2013-09-27 21:14:01,873:
com.tideway.integrations.vsphere.ProviderSession:

INFO: returning ([-- 1: vSphereIntegrationResult
config_storageDevice_scsiLun["key-vim_host_ScsiLun-0005000000766d686261333a303a30"]_capabilities_updateDis
config_storageDevice_scsiLun["key-vim_host_ScsiLun-0005000000766d686261333a303a30"_deviceName : /vmfs/devi
vsphere_id : host-1867
vsphere_type : HostSystem
], {})

Displaying custom attributes
The BMC Atrium Discovery data model allows additional attributes to be added to nodes. You
should only ever add additional attributes to inferred nodes. You must never update DDD nodes.

BMC Atrium Discovery 10.2

Page 1631 of 1671


Attributes might be added by patterns at any point in time. It is strongly recommended that you add attributes at creation time when building your own inferred nodes. Inferred nodes that were created by the system must be modified using a separate pattern.

Once you have modified the node it is necessary to make the attribute visible. You can do this in the following ways:

- Using `model.addDisplayAttribute`
- Populating the `_tw_meta_data_attrs` attribute on the inferred node

**To display the custom attribute on the node view using `model.addDisplayAttribute`**

You can add the newly created attribute, or list of attributes, to those displayed on the node page using the `model.addDisplayAttribute` function. For example, to display the `server_name` attribute, use the following:

```c
if si.server_name then      model.addDisplayAttribute(si, "server_name");
```

Similarly, to display the attribute list `list_attrs`, use the following:

```c
list_attrs :=[]; // list of attribute names      if si.port then
```

**To remove the custom attribute from the node view**

You can remove attributes that you have created, or a list of attributes that you have created, from those displayed on the node page using the `model.removeDisplayAttribute` function. For example, to remove the `server_name` attribute, use the following:

```c
model.removeDisplayAttribute(si, "server_name");
```

Similarly, to remove the attribute list `list_attrs`, use the following:

```c
model.removeDisplayAttribute(si, list_attrs);
```

**Populating the `_tw_meta_data_attrs` attribute on the inferred node**

If using this method, you must ensure only one pattern modifies the `_tw_meta_data_attrs` value on a particular node otherwise the conflicting changes are likely to result in the loss of one of the updates. Using the `model.addDisplayAttribute` and `model.removeDisplayAttribute` functions is simpler.

**To add a custom attribute at node creation**

Adding a custom attribute when a node is created is shown in the example below.

```c
jrun_si := model.SoftwareInstance(key := key,
     name := name,
     type := type,
     version := full_version,
```
product_version := product_version,
build := build,
server_id := server_id,
_tw_meta_data_attrs := ['server_id']);

To add a custom attribute after node creation
You can also add a custom attribute to a node after it is created. This might be in the pattern that creates the node, or where you are adding an attribute to a system created node, in a separate pattern.

if server_name then      si_node._tw_meta_data_attrs := ['server_name']

To add a list of attributes
The following example code shows how to add a list of attributes to an existing node. The code creates a list, adds values, and assigns it at once.

ltw_meta_data_attrs :=[]; // local list of attribute names      if si.port then

si._tw_meta_data_attrs := ltw_meta_data_attrs;
Troubleshooting

This section provides details of possible problems and how to solve them.

- If you encounter a problem
- Collecting diagnostics
- Frequently Asked Questions
- Troubleshooting clusters
- Troubleshooting Windows proxies
- Writing efficient regular expressions
- Diagnosing Hostname Problems
- Unmounting partitions
- Appliance does not have an IP on eth0
- Support information

If you encounter a problem

If you encounter a problem while using BMC Atrium Discovery:

- Read the Release Notes for this release. Your problem might be a known issue.

If you still cannot solve the problem, log into the Customer Portal on the BMC website. http://www.bmc.com/support

We will investigate your issue and get back to you as soon as possible. We will record details in our defect tracking system and allocate a customer case number.

Viewing information about the appliance

In the case of a problem, you might need to provide details of your appliance to Customer Support.

To view appliance information

1. From the Appliance section of the Administration tab, select Appliance Configuration. The Identification tab of the Appliance Configuration page is displayed.

   The fields are described in the Table below.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the appliance. By default this is Discovery_Appliance. This name is displayed at the top-left side of the BMC Atrium Discovery User Interface and in the title bar of the browser.</td>
</tr>
<tr>
<td>ID</td>
<td>The appliance ID.</td>
</tr>
<tr>
<td>Admin Email</td>
<td>The email address of the appliance administrator.</td>
</tr>
<tr>
<td>Support Email</td>
<td>The email address of Customer Support.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Description</td>
<td>A free-text description of the appliance.</td>
</tr>
<tr>
<td>Serial Number</td>
<td>The serial number allocated to the appliance. This uniquely identifies the appliance.</td>
</tr>
<tr>
<td>Commission Date</td>
<td>The date that the appliance was commissioned.</td>
</tr>
<tr>
<td>Kickstart Version</td>
<td>Kickstart is a system configuration program. When an appliance is kickstarted the version of the kickstart configuration file used during installation is displayed in this field.</td>
</tr>
<tr>
<td>Software Version</td>
<td>The BMC Atrium Discovery software version of the appliance.</td>
</tr>
<tr>
<td>Software Revision</td>
<td>The RPM (Red Hat Package Manager) version for the appliance.</td>
</tr>
</tbody>
</table>

To view logs

1. From the Appliance section of the Administration tab, select Logs.
2. Select the Log files tab (by default, the Log files tab is displayed).
3. A list of all the available logs is displayed on the Appliance Logs page.
4. Click the Watch link next to any log to view the log in detail.

The logs are described in Contents of the logs.

To view session logs

You can show or hide session logs for individual hosts from the Host view window.

1. From the Summary section on the Infrastructure page, click Hosts.
2. Select a host from the list. The host's details are displayed.
3. Click the Session Logs icon on the dynamic menu. The dynamic drop-down is displayed with a list of session logs for this host. Session logs (and therefore the icon) will only exist if Session Logging was enabled in the credential when the host was scanned.

This screen illustrates the Session Logs for hosts.
4. Click the Session Logs link to show the list of session logs for that host. The Session Logs icon is not displayed if there are no logs associated with the host you have selected.

5. Select a session log to view and click it. The session log is displayed.

Contacting Customer Support

BMC Software, Inc.
5th Floor
157-197 Buckingham Palace Road
London
SW1W 9SP
United Kingdom

Tel: +44 (0) 800 279 8925
Email: customer_support@bmc.com
Web: http://www.bmc.com/support

Collecting diagnostics

The following topics provide information about gathering BMC Atrium Discovery diagnostics data:

- Collecting additional data for support cases
- Getting record data
- Contents of the logs
- Changing log levels at runtime

Collecting additional data for support cases

When you raise an issue with Customer Support, a support case is initiated that describes the observed behavior. This case is used to track the resolution of the problem. Frequently you will be asked for information about the state of the appliance, configuration information, logs, record data, and so on.

1. **Save time! Send the right information when opening the support issue**

   In case of problem related to an SNMP device, collect the device capture first.
   If you are not satisfied by the scan result of a host, collect:
   - the session logs
   - the record and pool data (see below)
   - the discovery and reasoning logs (see below)
Collecting additional data

You can collect additional data that Customer Support requests from the Appliance Support page (Administration > Appliance Support).

Based on whether you are using a standalone machine or a cluster, the way you gather additional data differs.

- On a standalone machine: You can directly access the Appliance Support page and collect additional data from the machine itself.
- On a cluster: You can collect additional data of the entire cluster, that is, cluster coordinator and the members of the cluster from the Appliance Support page of any member or the coordinator.

From the Appliance Support page you can do the following:

- Collect BMC Atrium Discovery logs between specified dates.
- Collect Target Host Data and session logs for specified IP addresses or ranges.
- Collect Miscellaneous files.
- Create named archives of collected data.
- Download and manage archives of collected data.

The following systems and aspects of the BMC Atrium Discovery application create logs which might be useful to Customer Support when diagnosing problems:

- **AppServer**: used to diagnose UI errors.
- **Security**: used to track user access and actions.
- **Performance**: used to diagnose performance problems.
- **Discovery**: used to diagnose discovery and data quality issues.
- **Model**: used to diagnose queries or reporting problems.
- **Reasoning**: used to diagnose discovery and data quality issues.
- **Cluster Manager**: Used to diagnose cluster problems.
- **Integrations**: used to diagnose problems with integrations.
- **Others**: used to diagnose miscellaneous problems, for example, script problems.
- **Phurnace provider**: used to diagnose web server discovery problems.
- **Mainframe provider**: used to diagnose mainframe computer discovery problems.
- **AppServer Errors** used to diagnose UI errors.
- **Audit Logs** used for general diagnostics and user errors.
- **Windows proxies**: used to diagnose Windows discovery problems.

The discovery data and logs are also stored on the appliance. These might help Customer Support diagnose discovery problems. Enter the IP Ranges for which you wish to collect data:

- **Get session logs**: raw discovery data used for diagnosing discovery and data quality issues.
- **Record data**: used to reproduce discovery problems on test appliances.
- **Pool data**: used to reproduce discovery problems on test appliances.

The following files and system logs might be useful to Customer Support when diagnosing problems with the appliance:

- **Configuration files**: used to check the appliance configuration.
- **Customer extensions**: used to check customer configured extensions.
- **Export extensions**: used to configure customer configured export extensions.
- **System messages**: used to diagnose appliance related problems.
- **omniNames files**: used to check communications configuration.
- **Installed packages**: used to ensure that the correct packages are installed.
- **bash history**: used to check the command line history.
- **sar logs**: used to diagnose performance problems.
- **Tripwire reports**: used to view these reports.
- **Atrium Discovery output files**: used to check the output files.
- **Generated rules**: used to check the generated rules.
- **Report usage**: provides information on the type of reports used and the number of times they were used.
- **Tomcat output file**: used to check the Tomcat configurations.
- **Windows proxy configuration files**: used to check the configuration of Windows proxies.

**Creating an Archive of Collected Data for Customer Support**

**To Select BMC Atrium Discovery Logs**

From the BMC Atrium Discovery Logs By Date section of the Appliance Support page:

1. Select the start date from which you want to collect logs using the drop-down selectors in the From field.
2. Select the end date using the drop-down selectors in the To field.
3. Using the check boxes, select the log files that Customer Support requested.

**To Select Target Host Data**

From the Target Host Data section of the Appliance Support page:

1. Enter one of the following in the IP Ranges field:
   - IPv4 address (for example 192.168.1.100).
   - IPv4 range (for example 192.168.1.100–105, 192.168.1.100/24, or 192.168.1.*) up to a maximum of 256 addresses.
   - IPv6 address (for example fe80::655d:69d7:4bfa:d768).
   - A comma separated list of ranges (IPv4 only) and addresses (for example 192.168.1.100, 192.168.2.*, fe80::655d:69d7:4bfa:d768).
2. To retrieve session logs for these IP addresses, select the Get session logs check box.
3. Using the selection boxes, select the additional data files that Customer Support has requested.

**Gathering data where fully qualified domain names are important**

If fully qualified domain hostnames (FQDNs) are important where the discovery target addresses include IPv6 addresses, then every IPv6 address that appears in the interface list on the root node (Host, NetworkDevice or SNMPManagedDevice) details page needs to be separately specified in the IP address list on the gather page (as well as any IPv4 addresses).

**To Select Miscellaneous Files**

From the Miscellaneous Files section of the Appliance Support page:

1. Select the start date from which you want to collect logs using the drop-down selectors in the From field.
2. Select the end date using the drop-down selectors in the To field.
3. Using the check boxes, select the files that Customer Support have requested.

**To Create an Archive**

To create an archive of the selected logs, data, and files; from the Create Archive section of the Appliance Support page:

1. Enter a name for the archive in the Name field.
2. Enter a description for the archive in the Description field.
3. Click the Gather button.

The selected logs, data, and files are gathered into an archive and saved on the appliance. The archive and information relating to the archive are displayed in a new row of the Gathered Data Files section.

The following information is displayed for each archive:

- **Name**: the name of the archive. Click this to download the gathered data file.
- **Description**: the description entered when the archive was created.
- **Date**: the date that the archive was created.
- **Size**: the size (in bytes) of the archive.
- **User**: the user who created the archive.
- **State**: the state of the archive. This can be: Creating or Complete.
- **Summary**: to see a list of the archive contents in a popup window, click **View**.
Retrieving Archives to Send to Customer Support

To retrieve an archive, from the Gathered Data Files section of the Appliance Support page, perform the following:

1. Click the name of the archive.
   A file download window specific to your browser type is displayed.
2. Save the compressed file containing the archived data to a location on your file system.

Based on whether you are using a standalone machine or cluster, the content of the compressed file is as follows:

- In a standalone machine, it contains the archived data in a compressed file.
- In a cluster, the file contains a separate compressed archive for each member, bearing the name of the member IP address or hostname (whichever was specified by the user when creating the cluster). Corresponding XML files list the contents of each member archive.

**Gathered data for Windows hosts**

The gathered record data for Windows hosts is available only in the compressed file of the member that performed the discovery.

**tw_health_check**

You might be asked by BMC Support to run the `tw_health_check` tool in the event of an investigation. It will gather some generic configuration, performance and status information helpful to the support process.

**Getting record data**

The most useful information that you can provide customer support for the vast majority of cases that you raise is record data. Record data is a recording of the data returned during a discovery run structured so that it can be played back on another appliance, such that the appliance behaves as if it were actually scanning the IP addresses. With this, the support engineer should be able to reproduce the errors that you are seeing on a BMC test appliance.

**Setting the appliance to create record data**

The appliance must be set to store record data. To do this:

1. On the Administration tab, click Discovery Configuration in the Discovery section. The Discovery Configuration page is displayed.
2. Scroll down to the Others section and using the Recording Mode drop-down, select Record.
3. Click Apply to set the appliance to create record data. The appliance will now create record data for discovery runs until it is taken out of record mode.
Setting the Windows proxy to create record data

For Windows discovery to store record data, the proxy must also be set to record mode. To do this:

1. From the secondary navigation bar on the Discovery tab, click **Credentials**.
2. Click **Devices**.
3. Click **Windows Proxies**.
   The Device Credentials page is displayed which contains the Windows proxy pools and the corresponding Windows proxies.
4. Click on the Windows proxy name that you want to set to record mode.
   The proxy management page is displayed.
5. In the Proxy Settings section, using the Recording Mode drop-down, select **Record**.
6. Click **Apply Proxy Settings** to set the proxy to create record data. The proxy will now create record data for any discovery it performs until it is taken out of record mode.

Performing a discovery run in record mode

All discovery runs performed when the appliance is in record mode create record data. All discoveries using a proxy in record mode create record data. Performing a discovery run when an appliance and proxy are in record mode is done in exactly the same way as "normal" discovery.

Once the appliance, and proxy (for a Windows discovery) have been put into record mode, you must rescan the IP address, addresses, or IP range to create record data.

Once you have performed a discovery run to create the record data, you should set the appliance and proxy back to normal, that is, recording mode **Off** using the drop-downs described in the previous procedures.

To get record data

1. On the **Administration** tab, click **Appliance Support** in the Appliance section.
   The Appliance Support page is displayed.
2. In the Target Host Data section of the Appliance Support page, enter the IP address, addresses, or range of addresses into the **IP Range(s)** field.
3. Select **Record data**. (For Windows hosts, also select **Pool data**).
4. In the Create Archive Section enter a name for the archive and a description.
5. Click **Gather** to create the archive.
This screen shows the Appliance support page completed to collect record data from IP range 192.168.0.10-15.

When the archive is created, a row representing it is displayed in the Gathered Data Files section.

This screen show the Gathered Data files section with an entry for the gathered data file for IP range 192.168.0.10-15.

6. Click the name link to download the data file.
7. Send the data file to Customer Support as part of your case.

The Appliance Support page is described more fully here.

Contents of the logs

As each BMC Atrium Discovery component and script runs, it outputs logging information. Logs are all stored in `/usr/tideway/log`. The files can be accessed directly from the appliance command line or in the log viewer in the user interface. Logs can be downloaded from the appliance through the Support Services administration page.

Logs are critical to understanding problems that might be encountered in the system. This section describes the most important log files, and how to understand the information contained in them.

- Viewing log files
- Log levels
- Most important logs
- Log roll-over
- Service “out” files
- UI Tracebacks

Viewing log files

To view logs:

1. From the Appliance section of the Administration tab, select Logs.
2. Select the Log files tab (by default, the Log files tab is displayed).
3. A list of all the available logs is displayed on the Appliance Logs page.
4. Click the Watch link next to any log to view the log in detail.

The log level associated with each message is displayed. For information on log levels see Changing log levels at runtime.

⚠️ If the Watch link is not available the log is not present.

1. To delete a log, click the Delete link next to a log.
2. To download a log, click the Download link next to a log. You can then choose to open or save the text document for that log.
3. Click the Compress link next to a log for larger files that you want to download. The log is given a .gz extension and is described as 'Compressed' in the Type column. The Watch link is also removed for that log.

⚠️ You cannot uncompress a log once it has been compressed.

You can also delete old logs by clicking the Delete Old Logs button on the main log display page. An old log is one to which writing has been completed and there is a replacement currently being written.

Log levels

Log messages are classified into the following levels:

- **DEBUG** — fine-grained informational events that are most useful to debug the application.
- **INFO** — informational messages that highlight the progress of the application at coarse-grained level.
- **WARNING** — potentially harmful situations.
- **ERROR** — error events that might still allow the application to continue running.
- **CRITICAL** — severe error events that might cause the application to abort.
- **USEFUL** — start-up messages.

Unless asked to do otherwise by Customer Support, it is recommended that you configure each service to log at INFO level.

Most important logs

Each subsystem and script writes its own log file, with a name derived from the service or script. The ones that most often contain relevant information are:
Discovery log

tw_svc_discovery.log contains log messages from the Discovery subsystem, which is responsible for connecting to discovery targets and retrieving information from them. At INFO level, the log contains information about each discovery method that is executed; at DEBUG level, it contains a great deal of detail about how the discovery methods attempt different access mechanisms.

tw_svc_integrations_sql.log contains the log messages related to database scans.

Reasoning logs

The Reasoning subsystem coordinates discovery and executes patterns, maintaining the data model. On multi-cpu machines, the Reasoning subsystem is split into multiple “ECA Engine” processes; the log files are split correspondingly, so the files are suffixed _nn.log where nn is the engine number, such as 00 or 01.

Reasoning files are split as follows:

- tw_svc_reasoning.log contains information from the top-level Reasoning control process. It is periodically updated with information about the states of the active scanning ranges. In case of pattern activation error, this file will help you to find the root cause.

- tw_svc_eca_engine_nn.log contain information about scanning activities, including when each IP address starts and completes scanning. Each separate ECA engine process has its own log. On single CPU machines, the single ECA engine is part of the Reasoning control process, so this log output is in tw_svc_reasoning.log rather than a specific ECA engine log.

- tw_svc_eca_patterns_nn.log contain log messages from TPL patterns. If a pattern is misbehaving, its output will be here.

- tw_svc_eca_trackers_nn.log contain debug information about the event trackers in the ECA engines. They only contain useful output when the Reasoning subsystem is set to log at DEBUG level. The information is not intended for human consumption, but for processing by diagnostic tools by Customer Support.

Model log

The "model" subsystem contains the data store, search, taxonomy and audit services.

- tw_svc_model.log contains log messages from the model subsystem. At INFO level, it contains details of all searches, and information about database checkpoints. At DEBUG level, it contains a great deal of detail about how data is stored and retrieved.
Application server log
The main web user interface is hosted by an application server. tw_appserver.log contains log message from the application server, including information about pages viewed and any errors that can occur in rendering the UI.

Security log
tw_svc_security.log contains output from the Security subsystem, which is responsible for authenticating users and authorizing user actions. It contains information about authentication failures. If LDAP-based authentication is used, it also contains information about the state of the connection to the LDAP server.

Performance logs
performance.log.* contain information about the appliance load, obtained by running the top command every 10 minutes. This is useful for diagnosing performance issues.

Script logs
In addition to the main subsystem logs, the logs from a number of scripts are often relevant:

tw_upgrade.log contains a log of actions performed during a BMC Atrium Discovery version upgrade.

tw_tax_import.log contains a log of the actions of the Taxonomy importer. If a Taxonomy extension fails to import, the details will be in this log.

tw_scan_control.log contains any messages output by the tw_scan_control tool for creating scanning ranges.

tw_imp_csv.log contains messages output by the CSV import tool, which can be run from the command line as tw_imp_csv, or via the web UI in Administration > CSV Data Import.

CMDB Synchronization logs
tw_exporter_connection_test.log contains the logs generated by the CMDB sync test connection.

tw_svc_cmdbsync_transformer.log contains the logs generated by the syncmapping files.
tw_cmdbsync_exporter.log contains the rest of the logs.

⚠️ For clustered deployments, CMDB Syncronization logs are stored on the cluster coordinator appliance only.
Log roll-over

To prevent the log files growing without bound, they are rolled on a daily basis. The first time a log message is output after midnight (appliance time), the current .log file is renamed to correspond to the date, in the form .log. YYYY.MM.DD, and a new log file ending simply .log is opened.

As it is the output of a log entry after midnight that causes the roll-over, if a subsystem has not output any log messages since midnight, the .log file might contain entries from the previous day(s). This also means that unless execution of a script straddles midnight, script log files do not roll over, and the .log file will contain information about every run of the script.

Logs are automatically compressed seven days after they are created. When a log is compressed it is no longer available in any of the log selector drop down lists.

Compressed logs are automatically deleted 30 days after the initial log file was created.

Old logs can be manually compressed or deleted in the log viewer, or directly using the command line.

Service "out" files

Each running component in the system captures anything that is sent to stderr and stdout and logs it to a file. These files are all of the form <service name>.out and are also stored in /usr/tideway/log.

In normal usage, the *.out files generally contain just a single line to indicate that the service has successfully started. Another line is stored to indicate when the service is shut down cleanly. Any other content is often an indication that something has gone awry. However, there are some exceptions to this rule. For example, the tw_appserver.out files always have some content, as the appserver sends certain information to stdout. This means that you need to read the contents to check for any errors that might be in there.

The *.out files are rolled each time a component is started. The most recent file always ends just .out; the previous file ends .out.1, the next earliest .out.2 and so on. Nine previous .out files are kept, so the oldest file ends .out.9.

UI Tracebacks

Whenever the appserver component experiences an unhandled error, you are presented with an error message. When this happens the UI is obscuring a detailed traceback page which might contain useful information that can assist you with troubleshooting and debugging activities.

The appserver saves the entire html page that is being hidden as a self-contained html file that can be opened in a browser. These files are stored in the following location on the appliance:

/usr/tideway/python/ui/web/ErrorMsgs

You can manually delete these files from the Appliance command line.
Changing log levels at runtime

For each service that you have running in BMC Atrium Discovery, you can dynamically change the log level without having to restart the appliance.

To change log levels

1. Click the Administration tab from the primary navigation bar across the top of the page. The Administration home page is displayed. For a full description, see the Administration Page.
2. Click the Logs icon in the Appliance section. The Logs page is displayed showing the Log Files tab.
3. Click the Logging Levels tab. Each of the available service settings has a drop-down menu which enables you to change the log level for that setting without having to restart the appliance. The current logging level for each setting is also displayed for reference purposes.

The following log levels are available for selection:

- **Debug** — Specifies fine-grained informational events that are most useful to debug the application.
- **Info** — Specifies informational messages that highlight the progress of the application at coarse-grained level.
- **Warning** — Specifies potentially harmful situations.
- **Error** — Specifies error events that might still allow the application to continue running.
- **Critical** — Specifies severe error events that might cause the application to abort.

The log levels are cumulative in the order they are displayed in the drop-down. Therefore, the Debug log level includes everything including debug messages, Info includes everything up to and including the Info level, and so forth.

If you change the log levels on this page, the levels will persist. In other words, the next time that you restart the appliance the log level settings that you have selected will remain.

When you change one or more of the log levels from the drop-down menu against any setting, click **Apply**. A message is displayed informing you that the change you have made has been successful. This message will continue to be displayed until you select another page in BMC Atrium Discovery and return to the Logging window. The **Apply** button is not enabled until you have changed one or more of the log levels.

⚠️ Potential performance impact of Debug log level

If you select Debug for any setting a pop-up warning is displayed asking you to confirm your selection. This is because the Debug log level can impact the performance of the system.
Frequently Asked Questions

What is the performance impact of running BMC Atrium Discovery?

Running BMC Atrium Discovery should have a minimal impact on your environment. The discovery techniques used are non-intrusive, lightweight and agent-free.

What can BMC Atrium Discovery discover in the network?

BMC Atrium Discovery is IP-based and can discover any host system with an IP connection including servers, workstations, network nodes, printers, wireless access points, and so on. In actuality, though, we aim BMC Atrium Discovery at datacenter discovery, and it is optimised to that purpose. For this reason, we do not explicitly support more client-side items, such as wireless access points, workstations and so on. Any support for those that does exist is a side effect of our support for server-side discovery, and we’re unlikely to invest in improving it.

What discovery techniques do you use?

BMC Atrium Discovery uses a range of discovery techniques where appropriate. These include:

- Network scanning (looking for services on well-known TCP and UDP ports on IP-reachable machines).
- Remote command execution (looking at specific processes running on each node, querying package managers and querying established inter-process communications mechanisms).
- SNMP (MIBs provide a rich source of management information).

Will any network security need to be disabled for the discovery process?

Obviously, the BMC Atrium Discovery appliance needs to be able to reach areas of the network in order to discover hosts. However various methods of providing secure access are possible without disabling firewalls and access control policies, including using VPN tunnels and using Windows proxy BMC Atrium Discovery appliances. Some IDS systems might identify certain activities (such as port scans) as suspicious.
What is the impact of my applications running on platforms that are not supported by BMC Atrium Discovery?

The discovery process will identify endpoints on such computers if they are visible from other hosts. You will need to complete details of programs running on them manually, though it might also be possible to categorize some of the components of the applications running on the unsupported platform either by which port it, or its counterpart, is listening on.

Can the product introduce any risk into my network or application infrastructure?

To provide a clear picture of your total IT infrastructure, BMC Atrium Discovery will actually reduce risk in your network by allowing you to weed out rogue elements that do not meet corporate policy, are out of date or provide potential security holes.

The BMC Atrium Discovery discovery process uses standard techniques that should not destabilize elements of the infrastructure.

Since there are always risks with deploying new technology, BMC's implementation plan involves analyzing areas of potential risk and achieving the right balance of risk and reward. BMC's test plan is also aimed at minimizing risk, ideally including testing in the customer's test environment.

Do I need to install any software on other computers?

The BMC Atrium Discovery ethos is agent-free management. BMC does not believe the logistical challenges associated with having an agent on every node is justifiable, so no BMC Atrium Discovery-specific software needs to be installed on other computers. The BMC Atrium Discovery user interface is entirely web-based.

Why is agent-free discovery so important?

Agent-based discovery relies upon a level of control of asset deployment that does not exist in most businesses. It also implies a significant cost overhead to maintain agents on each platform, including approving, testing and deploying the agents. Finally, agents might not be available for the range of target platforms that your organization uses. We use standard techniques that have individually been authorized and deployed.

How do I reset the BMC Atrium Discovery user password?

If you forget your user interface (UI) password to log in to BMC Atrium Discovery, you can reset the password at the command line.

The `tw_passwd` utility enables you to change the password of a specified user interface user. To use the utility, enter the following command at command prompt:

```
tw_passwd username
```
where:

- **username** is the name of the UI user to change.

For example:

```
[tideway@DE-32 ~]$ tw_passwd fred
New password:
Retype password:
Password set for user 'fred'.
[tideway@DE-32 ~]$
```

⚠️ **Changing passwords for command line users**

The `tw_passwd` utility is for changing UI users' passwords. To change the passwords for command line users, as the root user, use the Linux command `passwd`. This is described in [Changing the root and user passwords](#).

If you have any other questions about BMC Atrium Discovery, contact [Customer Support](#).

### Troubleshooting clusters

When you encounter problems with a cluster, the first thing that you see may be errors in the Cluster Manager UI. If you are unable to recover using the UI, you may be able to use the `tw_cluster_control` utility to fix the problem.

#### My cluster will not start!

When a cluster starts, one of the checks it performs is to ensure that the members are all the correct machines, rather than a different machine on the same IP address. For virtual machines, it checks the VM UUID. If the VM UUID has changed, the machine is assumed to be a different machine, and the cluster cannot start and logs the following critical error in `tw_svc_cluster_manager.log`.

```
Clustered machine: VM UUID has changed
Replace expected VM UUID by running: tw_cluster_control --replace-vm-uuid
```

The critical error is also displayed on the console if you are attempting to start `tw_svc_cluster_manager` manually.
To replace the VM UUID:
You can do this by running `tw_cluster_control` from the machine on which the cluster manager has failed.

1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` replace the expected VM UUID with the current value and enable the cluster service to start. Enter:

```
[tideway@wilkoapp1 ~]$ tw_cluster_control --replace-vm-uuid
[tideway@wilkoapp1 ~]$
```

3. Start the cluster manager. Enter:

```
[tideway@wilkoapp1 ~]$ sudo /sbin/service appliance start
[tideway@wilkoapp1 ~]$
```

4. Start the remaining services. Enter:

```
[tideway@wilkoapp1 ~]$ tw_cluster_control --cluster-start-services
[tideway@wilkoapp1 ~]$
```

To determine the health of a cluster
You can do this from any running member of the cluster.

1. Log in to the appliance command line as the tideway user.
2. Query the status of the cluster. Enter:

```
[tideway@wilkoapp1 ~]$ tw_cluster_control --show-members
```

This is an example of the cluster health check results for a three machine cluster with fault tolerance enabled where a non-coordinator machine cannot be contacted.

```
Cluster UUID : 508a243177476a01543289485ecb04e5
Cluster Name : Harness Cluster
Cluster Alias : 
Fault Tolerance : Enabled
```
Replication Factor : 2
Number of Members : 3

<table>
<thead>
<tr>
<th>UUID</th>
<th>Harness Cluster-01</th>
<th>Harness Cluster-02</th>
<th>Harness Cluster-03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Harness Cluster-01</td>
<td>Harness Cluster-02</td>
<td>Harness Cluster-03</td>
</tr>
<tr>
<td>Address</td>
<td>wilkoappl1.tideway.com</td>
<td>137.72.94.205</td>
<td>wilkoapp3.tideway.com</td>
</tr>
<tr>
<td>Cluster Manager Health</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_HEALTH_ERROR Communication failure</td>
</tr>
<tr>
<td>Overall Health</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_HEALTH_OK</td>
<td>MEMBER_HEALTH_ERROR Communication failure</td>
</tr>
<tr>
<td>Services</td>
<td>SERVICES_RUNNING</td>
<td>SERVICES_RUNNING</td>
<td>SERVICES_UNKNOWN</td>
</tr>
<tr>
<td>State</td>
<td>MEMBER_STATE_NORMAL</td>
<td>MEMBER_STATE_NORMAL</td>
<td>MEMBER_STATE_NORMAL</td>
</tr>
<tr>
<td>Coordinator</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>CPU Type</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
<td>Intel(R) Xeon(R) CPU E5620 @ 2.40GHz</td>
</tr>
<tr>
<td>Processors</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Memory</td>
<td>3833M</td>
<td>3833M</td>
<td>3833M</td>
</tr>
<tr>
<td>Swap</td>
<td>8191M</td>
<td>8191M</td>
<td>8191M</td>
</tr>
<tr>
<td>Free Space</td>
<td>/usr 33014M/38458M (15%)</td>
<td>/usr 33077M/38458M (14%)</td>
<td>/usr 32671M/38458M (16%)</td>
</tr>
</tbody>
</table>

[tideway@wilkoappl ~]$
Removing all failed machines from the cluster

You can do this by running `tw_cluster_control` from any running member of the cluster. The order of these steps is particularly important. See the note below.

1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` to remove all failed machines from the cluster. You need to supply the password of the system user. Enter:

   ```bash
   [tideway@wilkoapp1 ~]$ tw_cluster_control --remove-broken
   Password:
   Found 1 broken member:
   Harness Cluster-03 (wilkoapp3.tideway.com)
   Are you sure you want to make these changes? (y/n) y
   1 member is being removed
   [tideway@wilkoapp1 ~]$
   ``

3. Before you can reuse any machine removed from the cluster, you must revert it to a standalone configuration. All data is lost in this process. On the command line of the machine you want to revert, enter:

   ```bash
   tideway@wilkoapp3 ~]$ tw_cluster_control --revert-to-standalone
   Removing this machine from the current cluster
   WARNING: This will delete all data on this cluster member!
   Do you wish to proceed? (yes/no) yes
   Stopping Cluster Manager service:
   Stopping Cluster Manager service: [ FAILED ]
   Removing cluster configuration
   Starting Cluster Manager service:
   Starting Cluster Manager service: [ OK ]
   Removing appliance configuration
   Deleting datastore
   Starting Security service: [ OK ]
   ...
   ...
   Starting Reasoning service: [ OK ]
   Importing TKU packages: [ Unknown ]
   Writing TKU-Core-2064-02-1-ADDM-10.0+: [ OK ]
   [ TKU-Extended-DB-Discovery-2064-02-1-ADDM-10.0+: [ Unknown ]
   [ TKU-Extended-Middleware-Discovery-2064-02-1-ADDM-10.0+: [ Unknown ]
   [ TKU-System-2064-02-1-ADDM-10.0+: [ Unknown ]
   ```
If a machine in the cluster has failed, and you `--revert-to-standalone` before removing it from the cluster, then you must shut down the reverted machine before issuing the `--remove-broken` command.

### A machine has failed in my non-fault tolerant cluster

A cluster that has no fault tolerance cannot survive when a machine fails. All data is lost and the only remaining option is to revert all of the machines to a standalone state and then create a new cluster. If you have backed up your cluster, you can restore the backup onto a new cluster of the same size.

To revert all of the machines to a standalone state:

1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` to revert the machine to a standalone state. Enter:

   ```bash
   $ tw_cluster_control --revert-to-standalone
   Removing this machine from the current cluster
   WARNING: This will delete all data on this cluster member!
   Do you wish to proceed? (yes/no) yes
   Stopping Cluster Manager service:
   ...
   ```

3. Repeat for all members of the cluster.

### Restoring a failed and forcibly removed coordinator

If a coordinator fails and is forcibly removed from the cluster, it must be reverted to a standalone state before it can be used again. To do this:

1. Log in to the appliance command line as the tideway user.
2. Use `tw_cluster_control` to revert the machine to a standalone state. Enter:

```
[tideway@wilkoapp1 ~]$ tw_cluster_control --revert-to-standalone
Removing this machine from the current cluster

WARNING: This will delete all data on this cluster member!

Do you wish to proceed? (yes/no) yes
Stopping Cluster Manager service:
...
```

The machine can be added back into the cluster.

Troubleshooting Windows proxies

If you have installed a Windows proxy and it fails to connect, there are a number of checks that you should make before contacting Customer Support.

Checking the keys and certificates

The appliance and Windows proxy communicate through an encrypted channel. In order for the Windows proxy to communicate with the appliance, each end requires the certificate of the other end. Visit the appliance user interface in `Discovery > Credentials > Windows Proxies` to see status information from the appliance point of view, and visit the Proxy Manager `Edit > Known Appliances` dialog. Ensure that both ends have correct certificate fingerprints and have approved the connection.

Checking for an omniORB.cfg file

The communication between appliance and Windows proxy uses the omniORB CORBA implementation. Configuration of omniORB can affect communication. omniORB's configuration is stored in `C:\omniORB.cfg`. In a normal Windows proxy install, that file is not present. If it is present, its contents might prevent communication with the Windows proxy, or might cause other problems.

Restricting appliances that can connect

omniORB can be configured to only permit connections from particular IP addresses. This is achieved by adding lines of this form in `C:\omniORB.cfg`:

```
serverTransportRule = <ip address> ssl
serverTransportRule = localhost tcp
```
Add one line for each appliance IP address with the ssl qualification, and one single line for localhost tcp. If multiple appliances are to use the same Windows proxy, one serverTransportRule line for each appliance should be added.

Only the specified IP addresses can connect with SSL, and internal communication between components of the Windows proxy is permitted over TCP.

Writing efficient regular expressions

BMC Atrium Discovery allows users to enter regular expressions (regexes) for a number of purposes, such as:

- associating TPL rules with events
- processing information in the body of a TPL pattern
- filtering out sensitive information from command lines.

These regular expressions are often matched against a large data set, so it is important that they are written to be efficient.

Why is regular expression efficiency important?

While a well written regular expression can be very efficient, a badly written regular expression might take a long time to run and slow the system down significantly. It is quite possible to write a regular expression that will take hours or days to finish - it is even possible to write a regular expression that will not finish within the lifetime of the universe when run against moderately sized strings.

Several improvements have been made in BMC Atrium Discovery to make it more robust against inefficient regular expressions than previous versions. It now minimises the regular expression matching needed when deciding which TPL patterns to run. It also spreads the work of running TPL patterns among multiple processors so that if one is busy with a long regular expression match, the other processors can carry on working.

Despite the improvements, writing efficient regular expressions is still important for keeping BMC Atrium Discovery running at its best. If BMC Atrium Discovery slows down significantly while scanning a network and either the reasoning or discovery service are using 100% CPU for long periods then one possible cause is an inefficient regular expression.

Anatomy of an inefficient regular expression

So how do you write an inefficient regular expression? One problem is when the regular expression does excessive backtracking; this can happen when there is more than one repetition operator in the regular expression. A repetition operator is +, *, or \{n,m\}. If the regular expression makes a partial match but then fails, then it must loop back and try any other possible partial matches in case any of them succeed.
For example, consider matching the regular expression $a.*b.*cd$ against the string $abc$ $abc$ $abc$. The match will never succeed since there is no $d$ in the string, but the regular expression must still check every possible combination of the letters $a$, $b$, and $c$ before it gives up. That is:

"*abc* abc abc",
"*ab*c ab*c* abc",
"*ab*c abc ab*c*",
"*a*bc a*bc* abc",
"*a*bc a*b*c ab*c*",
"*a*bc a*b*c ab*c*",
"*a*bc abc a*bc*",
"abc *abc* abc",
"abc *ab*c ab*c*",
"abc *a*bc a*bc*",
"abc abc *abc*"

As a rough guide the number of comparisons that the regular expression needs to perform is proportional to the length of the string times the number of possible intermediate matches.

In this example using the non-greedy operators, that is, $a.*?b.*?cd$, makes no difference to the number of matches it will make, since the regular expression engine still needs to try every combination.

**Real World examples**

Let's take a look at some examples based on real regular expressions that have caused problems in BMC Atrium Discovery:

`\b.*xx.*foo`

This was a regular expression that was compared against the command line of processes found on a host. It was failing when run against a half-megabyte string which included lots of repetitions of `xx` but did not contain `foo`.

Let's break down what happens when it is matched against a string:

1. The engine starts scanning from the start of the string.
2. The engine scans forward until it finds a word boundary `\b`.
3. The engine scans forward from the word boundary until it finds a matching `xx`.
4. The engine scans forward from the `xx` until it finds the string `foo` or it reaches the end of the string.
5. If it has reached the end of the string and not matched \texttt{foo}, it loops back to step 3 and scans forward to the next \texttt{xx}.

6. If it has matched all the \texttt{xx} and still not found \texttt{foo}, it loops back to step 2 and scans forward to the next word boundary.

So the regular expression matching contains nested loops; the total processing time is determined by the length of the string (for the command line that was causing the problem this was approximately 500,000) times the number of \texttt{xx} substrings (approximately 500) times the number of word boundaries (approximately 80,000). This was roughly equivalent to scanning a string twenty trillion characters long, and took more than a day to complete.

This was fixed in two ways:

Firstly, the \texttt{\b.*} was removed from the start of the regular expression, since it served no purpose other than to slow the whole thing down. This change reduced the runtime from days to a few seconds.

Secondly, we can use knowledge about the data we want to match; in this case we are only interested in the text from the start of the string up to the first space. So to stop the regular expression scanning the entire string we can anchor the regular expression to the start of the string with \texttt{^} and use the \texttt{\S} token to match non-whitespace characters. The final regular expression \texttt{^\S*xx\S*foo} will stop as soon as it reaches a whitespace character. It now takes a few microseconds when run against the same string.

\textbf{\texttt{-A(\D+)+-B(\D+)}}

This was used as a sensitive data filter. The intention was to scan command lines and remove options that start with \texttt{-A} and \texttt{-B}. However it not only did not do what the writer intended, it performed in a manner that could potentially take forever to process.

Let's break it down:

1. Scan from the start of the string until we find \texttt{-A}.
2. Match all non-digit characters until we find \texttt{-B}.
3. If \texttt{-B} is not found then try matching every combination of groups between \texttt{-A} and the end of the string or the next digit. For example, if the remainder of the string was \texttt{abcd} then it would match each of the following groups for \texttt{(\D+) +}
   - (abcd)
   - (abc) \texttt{(d)}
   - (ab) \texttt{(cd)}
   - (ab) \texttt{(c) (d)}
   - (a) \texttt{(b) (cd)}
   - (a) \texttt{(bc) (d)}
   - (a) \texttt{bcd}
   - (a) \texttt{(b) (c) (d)}.  


The number of combinations will double for each additional character in the remaining string.

So for the situation where the command line contains \( -A \) but not followed by \( -B \) it will take a time proportional to \( 2^N \), where \( N \) is the number of characters between the \( -A \) and the next digit or end of the string. To put that into perspective, on a standard PC a string of 22 characters takes about one second. A string of 40 characters would take about 3 days, and a string of 100 characters would take 9,583,696,565,945,500 years, though this has not been tested.

This regular expression was fixed by removing the group repetition, since it served no purpose:

\[-A (\D+) -B (\D+) \]. The runtime went down from forever to a few microseconds.

**Guidelines for writing efficient regular expressions**

This section provides guidelines to help you avoid common problems and write efficient regular expressions.

**Consider the failure scenarios**

As the previous examples show, the problems occur when a regular expression fails to match completely, but there are multiple partial matches. When writing a regular expression it is not sufficient to consider what happens when the regular expression succeeds, but also how it performs when it fails. The regular expressions used in BMC Atrium Discovery often are matched against a large number of command lines which can be very long (up to a million characters is not unknown), and might contain text that matches parts of your regular expression but not the whole thing.

**Beware of multiple repetitions of wildcard tokens**

A wildcard token is anything that can match more than one character; this includes: ., \w, \D, character classes, negated character classes and so forth.

If you have a regular expression with two or more consecutive wildcard repetitions then there is the possibility of backtracking. For example, if the target string starts with \( a \), has a length of \( N \) characters, and does not contain an \( x \), then:

- \( a . * . x \) - will take \( N^2 \) matches.
- \( a . * x . * x \) - will also take \( N^2 \) matches, since \( x^* \) can be a zero-length match so can match anywhere in the string.
- \( a . * y . * x \) - will take \( N^* M \) matches, where \( M \) is the number of occurrences of \( y \).

**REALLY beware of nested group repetitions**

As described above, if there is a group that contains a repetition, and the group itself is also repeated, for example \( ( . * ) \) *, then the number of possible matches might increase exponentially.
Do not start a regular expression with wildcard repetitions

This is a special case of the second point above. The regular expression engines searches for a match anywhere in the string, so it tries to match the regular expression starting at the first character, then at the second character, and so on until it gets a match. A regular expression of \.*x is equivalent to a regular expression of ^.*?.*x which suffers from the backtracking problem described above.

Since this is a very common error, BMC Atrium Discovery looks for regular expression that start with the unnecessary \.* and strips it out where possible.

Only match what you really need

The regular expression should be designed to stop as soon as it has enough to match, or to know that it cannot match. For example consider the regular expression \b\w+XXX\.*

- the \b\w+ is redundant, and can be replaced by "\w". This will match in all the situations where the original regular expression matched.
- the \.* at the end is also redundant, since it will have no effect on whether the match will succeed or fail.

So the entire regular expression can be replaced with \wXXX.

The exception to this is in situations where you need to use the string that has matched, rather than simply test if a match has succeeded or failed.

Try to fail fast

Try to make the entire regular expression fail if it reaches a point where it could not possibly match the intended target.

An example of this is in the first real world example shown above. The regular expression ^\S*\S*foo will never scan a string past the first whitespace character, regardless of whether it succeeds or fails. In the context in which it was being used this meant the difference between scanning a sequence of perhaps a hundred characters, and scanning a sequence of hundreds of thousand characters.

Profile - especially the failure cases

It is important to test your regular expression to make sure that it matches what you expect it to. However it is also important to test it for performance against long strings that partially match your regular expression, for example a megabyte string of random characters.

Minimize the data you pull back from hosts

TPL patterns let you run commands on a host and retrieve data to be searched with a regular expression, for example for versioning information.

A common mistake is to pull back a huge amount of information, for example:
cat file1 file2 file3...

then running a regular expression on the data to extract one piece of information.

This can potentially return a lot of data, so it will not only take a long time for the regular expression to match, it will also be slow to transfer the data over the network and take up a lot of space in the datastore.

A better way is to only get the information that you are interested in by running commands on the host such as grep or head. For example:

grep VERSION file1 file2 file3...

You can then run the regular expression on the much shorter text that is returned.

Do not use groups unless absolutely necessary

When you enclose part of a regular expression in a group using parentheses, the regular expression engine must do extra work to save the text matched by the group in case it is needed later. This can slow the matching process down, in some cases by a factor of four or more.

If you need to use parentheses, for example for part of a regular expression that is repeated, but you do not need to use the contents of the group afterwards then you can use the non-grouping version of parentheses - (?:...). This is generally still slower than not having parentheses at all, but is faster than normal parentheses. For example:

- (abc|def) - slow. Only use if you need to use the matched text later.
- (?:abc|def) - faster
- abc|def - fastest

Think about the regular expression

While these guidelines might help, there is no substitute for the thought and understanding that comes from taking a step back re-examining your regular expression for efficiency and accuracy.

Regular expression optimisations for TPL triggers

BMC Atrium Discovery uses a regular expression index to improve pattern performance. The index is used to quickly eliminate those regular expressions that will obviously not be matched by a given string. This significantly reduces the number of regular expressions that must be executed. Consequently, reasoning can usually avoid executing the sort of pathological regular expressions that are described in the rest of this document.

When optimizing regular expressions for TPL pattern triggers it helps to have a basic understanding of how the index works.
The index

Regular expressions are indexed by a property called a hint. A hint is a case-insensitive string that is a substring of every string that matches the expression. For example, the expression `Hoist the (mizzen mast|main brace), ye (landlubbers|scurvy dogs)`! has three obvious hints:

- `{Hoist the }`
- `, ye "`
- `!`

For simplicity, each expression is indexed only by its longest hint. In the case of the example above, the hint would be `{Hoist the }`.

Some regular expressions do not have any hints. For example, `\d+[-/]\d+[-/]\d+` has no substring that must be part of the matching strings. The hint calculator is also (currently) fairly naive, and misses some hints that could potentially be extracted. Expressions that have no hints must be treated separately.

When trying to match a string, the index is queried for a set of the regular expressions whose hint appears as a substring of the string being matched. Once built, the index can perform this query very quickly. This set, combined with a set of expressions that have no hints, forms the set of all the regular expressions that can potentially be matched by the string. An attempt is made to match the string against each expression in turn until one is found or there are no expressions left, meaning that the string does not match.

Try to give a hint

Regular expressions with no hints must be run against every string given to the index. It is important, therefore, to try to use regular expressions for which hints can be calculated.

The index's hint extraction algorithm cannot cope with the following kinds of sub-expression and will use them to delimit hints:

- Built-in character classes such as `\d, \w, and \.`
- Custom character classes such as `[ijkxyz]`
- Optional expressions such as `a?` and `a*`
- Groups such as `(foo)+`
- Alternation such as `foo|bar`

Using alternation at the top level of the expression always prevents hints being extracted. Alternation inside a group does not prevent hint extraction from parts of the expression outside the group.

As a rule of thumb, fancy characters terminate hints.
Try to make unique hints

If your regular expression only has a hint like `java`, it will need to be checked against a lot of strings. Try to engineer your expressions so that their longest hints are fairly uncommon.

Further Reading

This section provides some suggested further reading.

**Mastering Regular Expressions by Jeffrey Friedl**

This O'Reilly book is the authority on regular expressions. Chapter 4 describes the mechanics of how regular expression engines match text, and Chapter 6 describes writing efficient regular expressions in greater depth than this short document.

**The Python regular expression Library Documentation**

The full specification of the regular expression syntax used in BMC Atrium Discovery can be found on the Python website.

Useful tools

There are a lot of useful tools for developing and testing regular expressions, and a few that have been found useful are listed below.

BMC Atrium Discovery uses the Python regular expression library. Other libraries might have different features, behavior and performance.

**Kodos — the Python regular expression debugger**

*Kodos* is a GUI for testing Python regular expression. It does not give timing information, but is useful for trying regular expression against test data and seeing what matches. It is open source and runs on Windows, Linux and Mac OS X.

**Ipython**

*Ipython* is an enhanced interactive python shell. It is useful for timing regular expression matches using the `timeit` command. For example:

```
In [1]: import re
In [2]: x = '-' + '1000000 + abc'
In [3]: timeit re.search('abc', x)
100 loops, best of 3: 2.28 ms per loop
In [4]:
```

Ipython is open source and runs on Windows, Linux and Mac OS X.
Regex Coach

Regex Coach is a GUI regular expression debugger similar to Kodos. The big advantage of Regex Coach is that you can step through the regular expression as it attempts a match and watch the backtracking in action. This is very useful for getting a deeper understanding of how regular expressions work, and for debugging regular expressions that take much longer than expected.

Regex Coach is written in Lisp but the regular expression library is similar to the Python library. The current version is available for Windows only but earlier versions are available for Linux and Mac OS X. It is free to use, but cannot be distributed commercially without permission.

Diagnosing Hostname Problems

Some issues are indicative of incorrectly configured hostnames.

Visualizations and export do not display

This problem can be identified if you see a 500 Error - Internal Server Error! page in the user interface (UI) when attempting to load visualizations. You will also see the following error message in $TIDEWAY/tomcat/logs/catalina.out:

```
java.net.UnknownHostException: local_hostname: local_hostname
```

where, local_hostname is the hostname set on the computer.

To resolve this problem, see the previous section, Setting the hostname locally.

Cannot access the UI — 500 internal server error

This error can have many causes, as it results from one or more of the tideway services failing to start. A common cause however is that localhost cannot be resolved. If this is the case errors of the following form are displayed when restarting the tideway services.

```
[current date/time] [error] couldn't resolve WKServer address
[current date/time] [error] Couldn't resolve hostname for WebKit Server
```

Unmounting partitions on used disks

To unmount a partition on a "Non ADDM Disk", log in to the appliance command line as the tideway user.

1. Check the mounted partitions. Enter:
In this example /dev/sdb2 is mounted on /mnt/old. /dev/sdb2 shows as a "Non ADDM Disk" in the disk configuration utility. It must be unmounted before it can be configured using the disk configuration utility where it is displayed as "New Disk".

2. Unmount the /mnt/old partition. Enter:

```bash
[tideway@appliance01 ~]$ sudo umount /mnt/old
[tideway@appliance01 ~]$
```

The /mnt/old partition is unmounted.

3. Check for enabled swap partitions. Enter:

```bash
[tideway@appliance01 ~]$ /sbin/swapon -s
Filename Type Size Used Priority
/dev/sda2 partition 4192956 0 -1
/dev/sdb1 partition 2928700 0 -2
[tideway@appliance01 ~]$
```

In this example /dev/sdb1 is an enabled swap partition.

4. Disable the /dev/sdb1 swap partition. Enter:

```bash
[tideway@appliance01 ~]$ sudo /sbin/swapoff /dev/sdb1
[tideway@appliance01 ~]$
```

This has an immediate effect on system resources.
Appliance does not have an IP on eth0

BMC Atrium Discovery and Dependency Mapping version 9.0 appliances require an IP address on eth0, either from a DHCP server, or as a static IP address. The following sections describe the troubleshooting steps that you must perform if the corresponding issues are reported:

eth0 is present but does not have an IP address

If an IP address on eth0 has not been configured, you must configure one by performing the following steps:

1. Log in to the appliance as the netadmin user.
2. From the Configure Network Interfaces menu, configure the eth0 IP address.
   For more information, see The netadmin user.

eth1 has an IP address instead of eth0

If the network interface is changed, the NIC might be provisioned as eth1 rather than eth0, in which case you must perform the following steps:

1. Log in to the appliance as the root user.
2. Delete the /etc/udev/rules.d/70-persistant-net.rules file. For example:
   rm /etc/udev/rules.d/70-persistent-net.rules
3. For the NIC to be provisioned as eth0, reboot the system.

Support information

This topic contains information about how to contact Customer Support and the support status for this and other releases.

Contacting Customer Support

If you have problems with or questions about a BMC product, or for the latest support policies, see the Customer Support website at http://www.bmc.com/support. You can access product documents, search the Knowledge Base for help with an issue, and download products and maintenance. If you do not have access to the web and you are in the United States or Canada, contact Customer Support at 800 537 1813. Outside the United States or Canada, contact your local BMC office or agent.

Support status

As stated in the current BMC Product Support Policy, BMC provides technical support for a product based on time rather than number of releases. To view the support status for this release, see the BMC Atrium Discovery and Dependency Mapping Support page.
Best practices

The following best practices provide information that you can use to secure BMC Atrium Discovery and Dependency Mapping (BMC Atrium Discovery), increase its effectiveness, and improve your understanding of the overall system.

Overview

BMC Atrium Discovery automatically discovers the hardware and software in your data center, determines configuration and relationship data, and maps applications to the IT infrastructure. The following is not an exhaustive list of documentation to understand BMC Atrium Discovery in its entirety. However, it contains information that will improve your understanding of the overall BMC Atrium Discovery system.

- Page: The Community Edition
- Page: BMC Atrium Discovery - what it does
- Page: Big Discovery
- Page: Business Service Management
- Page: Supported versions of BMC Atrium Discovery
- Page: Out of the box discovery capabilities

Security

There a number of actions you should take to ensure that your BMC Atrium Discovery is secure before you put it into production. The following documentation offers some common best practices that you should consider to keep your system secure and provides a brief overview of the security aspects of BMC Atrium Discovery. It is intended to provide network administrators with the information required to run BMC Atrium Discovery in their environment. It also provides the information required to enable security teams to verify that BMC Atrium Discovery is secure and does not compromise the security of their network.

- Page: Configuring Web authentication settings
- Page: Managing security policies
- Page: System communications
- Page: Information security
- Page: Configuring HTTPS settings
Planning

To ensure effective results, you are required to consider a number of factors while planning the implementation. For example, you must understand the deployment prerequisites and requirements, the way the system is used for scanning and consolidating data, the hardware requirements, the factors that affect performance, and so on. The following documentation is frequently used by customers while implementing BMC Atrium Discovery.

- Page: Windows proxy compatibility matrix
- Page: Factors affecting performance
- Page: Consolidation
- Page: Hardware requirements

Installing

BMC Atrium Discovery is supplied as a virtual appliance and a kickstart DVD image. The following documentation guides you through a typical BMC Atrium Discovery installation process. It contains important information, such as the installation prerequisites, required appliance specification, post-installation tasks, and so on. These recommendations are important for a successful installation of the system. However, prior to installing BMC Atrium Discovery, you must refer to and understand the Installation, migration, and upgrade overview documentation.

- Page: Installing the virtual appliance
- Page: Installing
- Page: Installing BMC PATROL on a BMC Atrium Discovery system

Integrating

Integration of BMC Atrium Discovery with BMC Atrium CMDB, Single Sign-On, and Orchestrator is supported. Integration of BMC Atrium Discovery with these products helps you keep your data up-to-date, use the BMC single sign-on and single sign-off mechanism, and automatically start a rescan when triggered by certain events. To understand how to integrate these products with BMC Atrium Discovery and what are the benefits of such integrations, see the following documentation.

- Page: Integrating with BMC Atrium Single Sign-On
- Page: BMC Atrium Orchestrator VMotion events
- Page: CMDB synchronization
Maintaining

To maintain and troubleshoot BMC Atrium Discovery, there are some common tasks that you may need to perform. The appliance baseline conditions are verified to get a health check of the system. In addition, some possible problems that you may encounter have been identified by BMC. The following documentation contains helpful information for maintaining and troubleshooting BMC Atrium Discovery.

- Page: Shutting down and restarting clusters
- Page: Shutting down and restarting machines in a cluster
- Page: Baseline configuration
- Page: Shutting down restarting and maintenance mode
- Page: Troubleshooting