Database Products for IMS Advisors and Toolkit User Guide

Supporting

Version 5.0.00 of Backup and Recovery Solution for IMS
Version 4.0.00 of BMC Fast Path Enhanced Online Suite
Version 4.0.00 of BMC Fast Path Offline Suite
Version 4.0.00 of BMC Fast Path Online Suite
Version 3.0.00 of MAXM Database Advisor for IMS
Version 5.0.00 of MAXM Reorg/EP Express for IMS
Version 5.0.00 of MAXM Reorg/EP for IMS
Version 5.0.00 of MAXM Reorg/EP for IMS with Online/Defrag Feature
Version 5.0.00 of MAXM Reorg for IMS
Version 5.0.00 of MAXM Reorg for IMS with Online/Defrag Feature
Version 5.0.00 of MAXM Reorg/Online for IMS
Version 5.0.00 of POINTER CHECKER PLUS

December 2015
Contacting BMC Software

Several methods are available for contacting BMC Software.

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

<table>
<thead>
<tr>
<th>Address</th>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC SOFTWARE INC</td>
<td>1 713 918 8800</td>
<td>1 713 918 8000</td>
</tr>
<tr>
<td>2103 CITYWEST BLVD</td>
<td>or</td>
<td></td>
</tr>
<tr>
<td>HOUSTON TX 77042-2827 USA</td>
<td>1 800 841 2031</td>
<td></td>
</tr>
</tbody>
</table>

Outside United States and Canada

<table>
<thead>
<tr>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td>+01 713 918 8800</td>
<td>+01 713 918 8000</td>
</tr>
</tbody>
</table>

© Copyright 2006-2015 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

Explorer is a trademark or registered trademark of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

DB2, IBM, IMS, RAA, RACF, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2103 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

Support website
You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support. From this website, you can:

- Read overviews about support services and programs that BMC offers
- Find the most current information about BMC products
- Search a database for problems similar to yours and possible solutions
- Order or download product documentation
- Download products and maintenance
- Report a problem or ask a question
- Subscribe to receive proactive e-mail alerts
- Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail
In the United States and Canada, if you need technical support and do not have access to the web, call 1 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:yourSupportContractID, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC
Have the following information available so that Customer Support can begin working on your issue immediately:

- Product information
  - Product name
  - Product version (release number)
  - License number and password (trial or permanent)
- Operating system and environment information
  - Machine type
  - Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - System hardware configuration
  - Serial numbers
  - Related software (database, application, and communication) including type, version, and service pack or maintenance level
- Sequence of events leading to the problem
- Commands and options that you used
- Messages received (and the time and date that you received them)
  - Product error messages
  - Messages from the operating system
  - Messages from related software
License key and password information

If you have questions about your license key or password, contact Customer Support through one of the following methods:

■ Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter SupID:yourSupportContractID, such as SupID:12345.)

■ In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.

Contents

About this book ............................................................... 21
Related publications ......................................................... 21
Conventions ................................................................. 22
Syntax diagrams ............................................................ 22
Summary of changes ...................................................... 24

Chapter 1 Getting to know Advisor and DBA Toolkit functions .................................................. 33
Information requirements ................................................. 33
Products that include console-related functions and features ................................................. 33
Summary of concepts ...................................................... 34
  Data management ......................................................... 35
  Traditional database maintenance cycle .................................... 35
  Advisor database maintenance cycle ...................................... 35
  Workflow with Advisor .................................................. 36
  Functions and features .................................................. 37
  Summary of benefits .................................................... 38
Data management .......................................................... 39
  Goals .............................................................................. 39
  Challenges ..................................................................... 40
  Strategies ...................................................................... 41
Traditional database maintenance cycle ......................................... 41
  Traditional Plan process ................................................. 42
  Traditional Gather process ............................................. 43
  Traditional Analyze process ........................................... 43
  Traditional Execute process ............................................ 43
Advisor database maintenance cycle ........................................... 44
  Advisor Configure process ............................................. 45
  Advisor Gather process ............................................... 46
  Advisor Analyze process .............................................. 46
  Advisor Execute process .............................................. 47
Advisor workflow .......................................................... 47
  Graphical user interface ................................................ 50
  Off-hour processing ..................................................... 51
  Exception list .............................................................. 53
  Problem resolution ...................................................... 54
  Statistical research ...................................................... 55
Using the console ................................................................. 77
Identifying IMS components ................................................. 78
  Defining your IMS environment ........................................... 78
  Defining database groups .................................................. 79
  Searching for databases ..................................................... 79
Customizing parameters and schedules ................................. 80
  Configuring JCL parameters .............................................. 80
  Editing parameters .......................................................... 80
  Editing schedules ........................................................... 81
Handling exceptions ............................................................ 81
  Working with the Exception List ......................................... 81
  Working with exceptions .................................................. 83
  Working with solutions .................................................... 83
  Analyzing a database exception ......................................... 83
  Using the Solution batch utility ......................................... 84
Controlling conditional processes ........................................... 84
  Controlling the IC Triggering by CA feature ......................... 84
  Controlling the Conditional Image Copy feature ................... 85
  Controlling the Conditional Reorg feature from the console ... 85
  Using the stand-alone Conditional Reorg utility .................... 86
Performing common database administration tasks ............... 87
  Generating JCL .............................................................. 87
  Generating recovery JCL .................................................. 88
  Viewing job output ........................................................ 89
  Using historical reports ................................................... 89
  Disassembling IMS control blocks, and mapping DBDs ........... 90
  Migrating to HALDBs and managing HALDB partitions .......... 90
  Tuning partitions and randomizers ..................................... 91
  Calculating space for a DEDB area ..................................... 91
  Annotating objects .......................................................... 92
Managing product components ............................................ 92
  Controlling user access ................................................... 92
  Managing repository data ............................................... 93
  Cloning repository data .................................................. 93

Chapter 3  Installing and using the console ............................... 95
Information requirements .................................................. 95
Verifying personal computer requirements ............................. 96
Configuring the console ....................................................... 96
  User requirements ........................................................ 96
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installing the console</td>
<td>97</td>
</tr>
<tr>
<td>Updating the console</td>
<td>103</td>
</tr>
<tr>
<td>Uninstalling the console</td>
<td>104</td>
</tr>
<tr>
<td>Launching and exiting the console</td>
<td>104</td>
</tr>
<tr>
<td>Managing BMC Database Management clients</td>
<td>105</td>
</tr>
<tr>
<td>Overview of the console</td>
<td>107</td>
</tr>
<tr>
<td>Navigation window</td>
<td>107</td>
</tr>
<tr>
<td>Message windows</td>
<td>108</td>
</tr>
<tr>
<td>Work area</td>
<td>109</td>
</tr>
<tr>
<td>Selecting user options</td>
<td>109</td>
</tr>
<tr>
<td>Setting up connections</td>
<td>110</td>
</tr>
<tr>
<td>Overview of the Enterprise List and personal connections</td>
<td>110</td>
</tr>
<tr>
<td>Managing the Enterprise List</td>
<td>110</td>
</tr>
<tr>
<td>Using the Add Host Connection command</td>
<td>111</td>
</tr>
<tr>
<td>Adding IMS objects</td>
<td>113</td>
</tr>
<tr>
<td>Chapter 4 Identifying and using IMS components</td>
<td>115</td>
</tr>
<tr>
<td>Defining your IMS database administration environment</td>
<td>115</td>
</tr>
<tr>
<td>Overview of IMS environment definitions</td>
<td>115</td>
</tr>
<tr>
<td>Defining IMS components by using the Auto Configure wizard</td>
<td>118</td>
</tr>
<tr>
<td>Defining IMS components manually</td>
<td>123</td>
</tr>
<tr>
<td>Working with database groups</td>
<td>126</td>
</tr>
<tr>
<td>Overview of groups and database ownership</td>
<td>126</td>
</tr>
<tr>
<td>Adding groups during the Auto Configure process</td>
<td>127</td>
</tr>
<tr>
<td>Adding groups manually from RECONs</td>
<td>127</td>
</tr>
<tr>
<td>Managing groups by using the Assigned Databases report</td>
<td>129</td>
</tr>
<tr>
<td>Managing groups by using the Unassigned Databases report</td>
<td>131</td>
</tr>
<tr>
<td>Deleting groups</td>
<td>132</td>
</tr>
<tr>
<td>Renaming groups</td>
<td>132</td>
</tr>
<tr>
<td>Managing database ownership</td>
<td>133</td>
</tr>
<tr>
<td>Searching for databases</td>
<td>136</td>
</tr>
<tr>
<td>Overview of the search feature</td>
<td>136</td>
</tr>
<tr>
<td>Specifying search criteria</td>
<td>136</td>
</tr>
<tr>
<td>Deleting search results</td>
<td>137</td>
</tr>
<tr>
<td>Chapter 5 Using Database Advisor</td>
<td>139</td>
</tr>
<tr>
<td>Overview of Database Advisor processes</td>
<td>139</td>
</tr>
<tr>
<td>Collect process</td>
<td>140</td>
</tr>
<tr>
<td>Analyze process</td>
<td>142</td>
</tr>
<tr>
<td>Solution process</td>
<td>146</td>
</tr>
<tr>
<td>Using Database Advisor to manage Reorg thresholds</td>
<td>147</td>
</tr>
</tbody>
</table>

*Database Products for IMS Advisors and Toolkit User Guide*
Establishing Reorg threshold parameters ................................................................. 148
Requesting Database Advisor processes on demand ................................................ 148
Requesting data collection on demand ................................................................. 149
Requesting analysis on demand ............................................................................. 149
Requesting database registration synchronization on demand ............................. 149
Recognizing Reorg threshold exceptions ............................................................... 150
  AREA DOVF UTILIZATION TOO HIGH exception ...................................................... 150
  AREA FRAGMENTATION TOO HIGH exception ......................................................... 150
  AREA I/O PROBABILITY TOO HIGH exception ......................................................... 151
  AREA IOVF SPACE LOW exception ...................................................................... 151
  AREA RAA UTILIZATION TOO HIGH exception ...................................................... 151
  AREA SDEP FREE SPACE TOO LOW exception ....................................................... 151
  AREA USED PERCENT EXCEEDED exception .......................................................... 152
  FSE ERROR exception ........................................................................................... 152
  HDAM RAA DEFINITION TOO LARGE exception ..................................................... 152
  HDAM RAA DEFINITION TOO SMALL exception ..................................................... 152
  HIDAM PRIMARY INDEX ERROR exception .......................................................... 152
  INCORRECT CI SIZE exception .............................................................................. 153
  INSUFF ALLOC FREE SPACE (LS) exception ......................................................... 153
  INSUFF TOTAL FREE SPACE (LS) exception .......................................................... 153
  INSUFFICIENT ALLOCATED FREE SPACE exception ............................................. 153
  INSUFFICIENT FREE BLOCKS exception ............................................................... 154
  INSUFFICIENT FREE SPACE IN RAA exception .................................................... 154
  INSUFFICIENT SPACE exception ........................................................................... 155
  INSUFFICIENT TOTAL FREE SPACE exception .................................................... 155
  I/O PROBABILITY TOO HIGH exception ............................................................... 155
  MAXIMUM OSAM ALLOC EXCEEDED exception .................................................... 156
  MAXIMUM VSAM ALLOC EXCEEDED exception .................................................... 156
  POINTER ERROR exception ................................................................................... 156
  PRILOG SIZE EXCEEDS THRESHOLD exception ..................................................... 156
  RAA TOTAL FREE SPACE TOO SMALL exception ..................................................... 157
  RECLAIMABLE FREESPACE EXCEEDED exception ............................................... 157
  RECLAIMABLE IOVF EXCEEDED exception .......................................................... 157
  RECLAIMABLE SPACE EXCEEDED exception ....................................................... 158
  RECOVERY NEEDED COUNT INVALID exception .................................................. 158
  TOO FEW DOVF CONTROL INTERVALS exception .................................................. 158
  TOO FEW EMPTY RAPS exception ........................................................................... 158
  TOO FEW IOVF BLOCKS exception ......................................................................... 159
  TOO MANY CA SPLITS exception .......................................................................... 159
  TOO MANY CI SPLITS exception .......................................................................... 159
  TOO MANY EXTENTS exception ............................................................................. 159
TOO MANY FRAGMENTED DATABASE RECORDS exception ........................................ 160
TOO MANY ROOTS IN OVERFLOW exception ......................................................... 161
TOO MANY SEGMENT SPLITS exception ................................................................. 161
TOO MANY SPLIT ACROSS BLOCKS exception ..................................................... 162
TOO MANY VOLUMES exception ......................................................................... 162
UNALLOCATED DATASET exception .................................................................... 162
UNINITIALIZED DATASET exception ..................................................................... 162
Analyzing Database Advisor exceptions .............................................................. 162
Accessing menu commands .................................................................................. 163
Reviewing reports .................................................................................................. 164
Reviewing analysis parameters and schedules ...................................................... 164
Reviewing base values .......................................................................................... 164
Using the Exception Forecast report ...................................................................... 168
Generating an Exception Forecast report ............................................................. 168
Resetting Exception Forecast report criteria ....................................................... 170
Auditing databases to ensure analysis .................................................................. 170
Reporting audit problems in a batch report ......................................................... 171
Reporting audit problems as exceptions ............................................................. 172

Chapter 6 Using Recovery Advisor ....................................................................... 173
Overview of Recovery Advisor processes ............................................................. 173
Analyzing RECON records and statistics against threshold values ................... 173
Reporting exceptions in the Exception List window ............................................ 174
Recommending solutions, and generating JCL .................................................... 174
Using Recovery Advisor to manage Recovery thresholds .................................... 174
Establishing Recovery threshold parameters ...................................................... 175
Executing Recovery Collect .................................................................................. 175
Requesting Recovery Collect on demand ............................................................. 176
Using a batch job to execute Recovery Collect .................................................... 176
Working with Recovery Advisor exceptions ....................................................... 178
BACKOUT NEEDED exception ............................................................................. 178
CA DATA SET NOT CATALOGED exception ......................................................... 179
CA RUN IN ERROR exception ............................................................................. 179
DBDS IN MULTIPLE DBDS GROUPS exception .................................................. 179
DBDS NOT IN ANY DBDS GROUP exception ...................................................... 180
DBDS NOT IN CHANGE ACCUM GROUP exception .......................................... 180
EEQE(S) FOUND exception .................................................................................. 180
GENMAX BELOW THRESHOLD exception .......................................................... 181
GENMAX EXCEEDED THRESHOLD exception .................................................... 181
GRPMAX BELOW THRESHOLD exception ............................................................ 181
GRPMAX EXCEEDED THRESHOLD exception ...................................................... 182
HALDB PARTITION NOT INITIALIZED exception ............................................... 182
IC DATA SET IN ERROR exception ............................................................. 182
IC DATA SET NOT CATALOGED exception ................................................ 183
IC RECOMMENDED BY CHANGE ACCUM exception ............................... 183
IMAGE COPY NEEDED exception .............................................................. 184
IMAGE COPY RECOMMENDED exception ............................................... 184
MIN # OF CA RUNS NOT AVAILABLE exception .................................... 185
MIN # OF IC'S NOT AVAILABLE exception ............................................. 185
MORE LOGS THAN THRESHOLD exception ........................................... 186
NO AREA DATA SET AVAILABLE exception ........................................ 186
NO AREA DATA SET DEFINED exception .............................................. 186
NO CA RUN WITHIN TIME RANGE exception ..................................... 187
NO IC WITHIN TIME RANGE exception ................................................ 187
NO SECIC WITHIN TIME RANGE exception ........................................... 188
NO SECLOG WITHIN TIME RANGE exception ..................................... 188
NO SECSLD WITHIN TIME RANGE exception ..................................... 189
NO VALID IMAGE COPY exception ......................................................... 189
PREOPEN NOT SET FOR AREA exception ........................................... 189
PRILOG ENTRY IN ERROR exception ..................................................... 190
PRILOG ENTRY NOT CATALOGED exception ........................................... 190
PRISLD ENTRY IN ERROR exception ..................................................... 190
PRISLD ENTRY NOT CATALOGED exception ........................................... 191
PROHIBIT AUTHORIZATION FOUND exception ..................................... 191
READ ONLY exception ........................................................................ 191
RECOVERY NEEDED exception .............................................................. 192
RECOVERY NEEDED COUNT INVALID exception ................................... 192
RECOVPD BELOW THRESHOLD exception .............................................. 192
RECOVPD EXCEEDED THRESHOLD exception ...................................... 193
SECLOG ENTRY IN ERROR exception ..................................................... 193
SECLOG ENTRY NOT CATALOGED exception ........................................... 193
SECSLD ENTRY IN ERROR exception ..................................................... 194
SECSLD ENTRY NOT CATALOGED exception ........................................... 194
Working with Recovery Advisor solutions .......................................... 194
Clearing Recovery Advisor exceptions from the repository .................. 195

Chapter 7     Using RECON Advisor functions 197
RECON management components ......................................................... 197
DBRC List History function ................................................................. 198
Using Advisor for RECON management .............................................. 199
Establishing parameters for RECON management .............................. 200
Parameters for setting RECON thresholds ........................................... 200
Parameters for handling RECON contention .................................................... 201
Parameter to request backups of RECON data sets ......................................... 202
Parameter to identify IDCAMS DEFINE control statements .......................... 203
Recognizing RECON threshold exceptions .................................................... 203
DSN SIZE EXCEEDS THRESHOLD exception ................................................. 203
INSUFFICIENT FREE SPACE exception ..................................................... 203
PRILOG SIZE EXCEEDS THRESHOLD exception ........................................... 204
RECON DISCARDED / UNAVAILABLE exception ............................................... 204
TOO MANY CA SPLITS exception .................................................................. 204
TOO MANY CI SPLITS exception .................................................................... 205
TOO MANY EXTENTS exception ..................................................................... 205
TOO MANY VOLUMES exception ................................................................... 206
Using the RECON UTILITIES solution ......................................................... 206
Replace mode JCL .......................................................................................... 206
Reorg All mode JCL ....................................................................................... 207
Using the RECON Reorg utility ....................................................................... 208
Modes of the RECON Reorg utility .................................................................. 208
Requirements for executing the RECON Reorg utility .................................... 209
RECON Reorg utility JCL ............................................................................... 209
Clearing RECON exceptions and resetting statistics ........................................ 210
Return codes of the RECON Reorg utility ....................................................... 210
SYSOUT information from the RECON Reorg utility ..................................... 211
Responding to a WTOR from the RECON Reorg utility .................................... 212
Generating JCL to prepare RECONs for disaster recovery ............................... 213
Using the DBRC List History function ........................................................... 214
Accessing the DBRC List History window ...................................................... 214
DBRC List History window ............................................................................ 215
Events in the DBRC List History Window .................................................... 217
List History - DBRC Command Interface window .......................................... 219
List - History Generate JCL window .............................................................. 220
Security for the DBRC List History function ................................................ 221
Removing a third-party library from the Advisor configuration ....................... 222

Chapter 8 Customizing parameters and schedules 225
Configuring JCL parameters .......................................................................... 225
Customizing parameters ................................................................................ 226
Parameter categories ..................................................................................... 226
Levels of parameters ..................................................................................... 227
Parameter inheritance ...................................................................................... 227
Editing parameters .......................................................................................... 229
Viewing parameter information ..................................................................... 230
Reorg Thresholds parameters ................................................................. 230
Recovery Thresholds parameters ........................................................... 242
RECON Thresholds parameters ............................................................. 247
IMS Status parameters ........................................................................... 250
Goals parameters ................................................................................... 253
E-mail / WTO / REXX EXEC parameters ............................................... 253
IC Triggering by CA parameters ............................................................ 256
Conditional IC parameters ..................................................................... 257
Conditional Reorg parameters ............................................................... 257
Collect interval parameters ..................................................................... 258
Analysis parameters ............................................................................... 260
Job Control parameters ........................................................................... 260
Unload parameters .................................................................................. 264
Reorg parameters .................................................................................... 266
Image Copy parameters ........................................................................... 268
Additional Fast Path parameters ............................................................ 272
Repository Management parameters ....................................................... 273
Recovery & Change Accum parameters .................................................. 276
Customizing schedules ........................................................................... 279
Overview of schedules ............................................................................ 279
Database Advisor schedules .................................................................... 279
Recovery Collect schedule ...................................................................... 280
E-mail Consolidation schedule ................................................................. 280
Repository management schedules ........................................................ 280
Summary of scheduled functions ............................................................ 281
Editing schedules .................................................................................... 285
Checking scheduled function completion codes ...................................... 287
Using program CPCTSCHD to initiate Advisor scheduled tasks ............. 289
JCL and control statements for program CPCTSCHD ............................... 289
Security for program CPCTSCHD ............................................................ 292
Configuring dynamic allocation models and matrixes ............................. 293

Chapter 9 Handling exceptions ............................................................... 295
Overview of exceptions ........................................................................... 295
Exception records .................................................................................... 295
Exception status types ............................................................................. 296
Exception types ....................................................................................... 296
Working with exceptions and the Exception List ..................................... 303
Opening the Exception List ..................................................................... 304
Reading the Exception List ...................................................................... 305
Filtering the Exception List by exception status types ............................ 305
Viewing exceptions for an object .........................................................308
Acting on exceptions for an object ....................................................308
Accessing details about objects on the Exception List ....................308
Exporting exception data .................................................................309
Receiving notification about exceptions ..........................................310
Recipients and types of notification messages .................................310
Frequency of notification messages ..................................................310
Controlling notification .................................................................310
Generating the Database Exception batch report ..........................311
JCL for the EXCEPLST command ..................................................311
Control Statements for the EXCEPLST command .........................312
Exception IDs and keywords for filtering by exception type ...........318
Return codes for the EXCEPLST command ..................................321
Database Exception report .............................................................321

Chapter 10 Working with solutions, JCL generation, and viewing jobs 323
Overview of Advisor solutions and JCL generation .......................323
Overview of the solution process .....................................................323
Overview of Advisor solutions .......................................................325
Overview of JCL generation ...........................................................330
Methods for working with Advisor solutions and generated JCL ....331
Parameters that influence JCL generation .......................................332
User-defined DD statements in generated JCL ...............................332
Using solutions from the Exception List .........................................333
Choosing the recommended solution ..............................................334
Choosing a solution to correct selected exceptions .......................335
Using the recommended JCL shortcut ............................................337
Generating JCL from the Generate JCL window .........................338
Generating optimized JCL .............................................................338
Working with generated JCL .........................................................339
Handling exceptions with the Solution batch utility .......................341
Coding JCL for the Solution utility ................................................341
Coding control statements for the Solution utility .........................343
Generating solution JCL on demand from the console ..................346
Interpreting return codes .............................................................347
Using GENJCL output .................................................................348
Using the SOLVE Processing Log report .....................................349
Replacing the DLIAJCLN utility with the BRIUBSOL utility ..........349
Generating JCL with the JCL Generation utility .........................350
Coding JCL for the JCL Generation utility ..................................350
Coding the GENJCL command and keywords ............................351
Generating JCL with the Utility Migration JCL Generator ........................................ 359
   Starting the JCL Generator ................................................................................... 361
   Setting the JCL generation parameters ............................................................... 361
   Generating JCL ....................................................................................................... 369
Generating and working with recovery JCL .......................................................... 370
   Recovery Wizard .................................................................................................... 370
   Workflow Status repository ................................................................................. 370
   Workflow Status window ..................................................................................... 371
   Generating recovery JCL ...................................................................................... 371
   Working with recovery requests ......................................................................... 372
Viewing jobs in the console ........................................................................................ 372
   Viewing jobs and refreshing job statuses ........................................................... 373
   Filtering jobs that are displayed .......................................................................... 374
   Viewing job output ................................................................................................ 376
Maintaining solutions ................................................................................................ 379
   Viewing solution information .............................................................................. 379
   Adding a user solution ........................................................................................ 381
   Deleting a user solution ....................................................................................... 383

Chapter 11  Controlling conditional processes 385
Using the IC Triggering by CA feature ......................................................................... 385
   Recommended usage of the IC Triggering by CA feature .......................... 387
   Implementing the IC Triggering by CA feature ........................................ 387
   Change Accumulation utility processing .......................................................... 388
   Image Copy utility processing ........................................................................... 389
   Exceptions and solutions related to the IC Triggering by CA feature .......... 390
Using the Conditional Image Copy feature ................................................................ 390
   Conditional image copy processing ................................................................. 392
   Conditional image copy considerations ........................................................ 393
   Implementing the Conditional Image Copy feature ........................................ 394
Using the Conditional Image Copy Simulation utility ................................................. 395
   How the ICPVSMIC utility works .................................................................. 395
   Coding JCL to execute the ICPVSMIC utility .............................................. 395
   Using the ICPVSMIC report ............................................................................. 396
Using the Conditional Reorg feature ........................................................................ 397
   How the Conditional Reorg feature works ....................................................... 398
   MAXM/EP utilities versus Classic database reorganization utilities .......... 398
   Enabling the Conditional Reorg feature with the console ....................... 399
Using the stand-alone Conditional Reorg utility ......................................................... 401
   Coding required JCL ....................................................................................... 401
   Coding the ADVISE command and keywords ........................................... 401
Language syntax ............................................................................................................... 501
  Free format .................................................................................................................. 501
  Mixed-case text ........................................................................................................... 501
  Comments .................................................................................................................... 502
  Paragraph labels ......................................................................................................... 502
  String expressions ...................................................................................................... 503
  Concatenation bars ..................................................................................................... 503
Variable processing ....................................................................................................... 503
  Simple variables ......................................................................................................... 503
  Recursive variables .................................................................................................... 504
  Array variables .......................................................................................................... 505
  Array variable dimension counts ............................................................................ 506
  Array variable deletion ............................................................................................. 506
  Variable substitution within JCL ............................................................................... 507
  System variables ........................................................................................................ 508
CALL statement ............................................................................................................. 509
  Syntax ......................................................................................................................... 510
  Examples ..................................................................................................................... 510
  Description .................................................................................................................. 510
CALL PROGRAM statement ........................................................................................ 511
  Syntax .......................................................................................................................... 511
  Examples ...................................................................................................................... 511
  Description .................................................................................................................. 511
DO statement .................................................................................................................. 512
  Syntax .......................................................................................................................... 512
  Examples ...................................................................................................................... 512
  Description .................................................................................................................. 513
GOTO statement ............................................................................................................. 514
  Syntax .......................................................................................................................... 514
  Examples ...................................................................................................................... 514
  Description .................................................................................................................. 514
IF statements .................................................................................................................. 514
  Syntax .......................................................................................................................... 515
  Examples ...................................................................................................................... 515
  Description .................................................................................................................. 515
JCL GEN statements ....................................................................................................... 516
  Syntax .......................................................................................................................... 516
  Example ....................................................................................................................... 517
  Description .................................................................................................................. 517
LET statement ................................................................................................................. 518
  Syntax .......................................................................................................................... 518
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examples</td>
<td>518</td>
</tr>
<tr>
<td>Description</td>
<td>519</td>
</tr>
<tr>
<td>MSG statements</td>
<td>519</td>
</tr>
<tr>
<td>Syntax</td>
<td>520</td>
</tr>
<tr>
<td>Examples</td>
<td>520</td>
</tr>
<tr>
<td>Description</td>
<td>520</td>
</tr>
<tr>
<td>SCRIPT statements</td>
<td>521</td>
</tr>
<tr>
<td>Syntax</td>
<td>521</td>
</tr>
<tr>
<td>Examples</td>
<td>521</td>
</tr>
<tr>
<td>Description</td>
<td>522</td>
</tr>
<tr>
<td>SEARCH statement</td>
<td>522</td>
</tr>
<tr>
<td>Syntax</td>
<td>522</td>
</tr>
<tr>
<td>Examples</td>
<td>522</td>
</tr>
<tr>
<td>Description</td>
<td>523</td>
</tr>
<tr>
<td>SORT statement</td>
<td>524</td>
</tr>
<tr>
<td>Syntax</td>
<td>524</td>
</tr>
<tr>
<td>Examples</td>
<td>524</td>
</tr>
<tr>
<td>Description</td>
<td>525</td>
</tr>
<tr>
<td>SUBSTR statement</td>
<td>525</td>
</tr>
<tr>
<td>Syntax</td>
<td>525</td>
</tr>
<tr>
<td>Examples</td>
<td>526</td>
</tr>
<tr>
<td>Description</td>
<td>526</td>
</tr>
<tr>
<td>TRACE statement</td>
<td>526</td>
</tr>
<tr>
<td>Syntax</td>
<td>526</td>
</tr>
<tr>
<td>Example</td>
<td>527</td>
</tr>
<tr>
<td>Description</td>
<td>527</td>
</tr>
<tr>
<td>TRANSLATE statement</td>
<td>528</td>
</tr>
<tr>
<td>Syntax</td>
<td>528</td>
</tr>
<tr>
<td>Examples</td>
<td>528</td>
</tr>
<tr>
<td>Description</td>
<td>528</td>
</tr>
<tr>
<td>JCL script examples</td>
<td>529</td>
</tr>
</tbody>
</table>
About this book

This book contains detailed information about the functions and features of the BMC database administration products for the IBM IMS environment that are available in the console. The information in this book is intended for database administrators (DBAs) who are the primary users of the product.

Like most BMC documentation, this book is available in printed and online formats. To request printed books or to view online books and notices (such as release notes and technical bulletins), see the support website at http://www.bmc.com/support.

**Note**

Online books are formatted as PDF or HTML files. To view, print, or copy PDF books, use the free Adobe Reader from Adobe Systems. If your product installation does not install the reader, you can obtain the reader at http://www.adobe.com.

The software also offers online Help. To access Help, press F1 within any product or click the Help button in graphical user interfaces (GUIs).

Related publications

From the BMC Support Central website, you can use the following methods to access related publications that support your product or solution:


- View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:
  - Documentation Center (primary center and secured center)
  - Support Central (at http://www.bmc.com/support/mainframe-demonstrations)
— **BMC Mainframe YouTube channel** ([https://www.youtube.com/user/BMCSoftwareMainframe](https://www.youtube.com/user/BMCSoftwareMainframe))

- View individual product documents (books and notices) within the “A – Z Supported Product List” ([https://webapps.bmc.com/support/faces/az/supportlisting.jsp](https://webapps.bmc.com/support/faces/az/supportlisting.jsp)).

Products with online interfaces also offer online Help via the F1 key or, for graphical user interfaces (GUIs), via a **Help** button.

---

**Tip**

If you prefer hardcopy documentation, you can order it from your BMC sales representative or from Support Central. Also, from Support Central you can subscribe to receive proactive e-mail alerts when BMC issues notices.

---

### Conventions

This document uses the following special conventions:

- All syntax, operating system terms, and literal examples are presented in this typeface.

- Variable text in path names, system messages, or syntax is displayed in italic text: `testsystestsys/instance/fileNametestsys/instance/fileName`

- Menu sequences use a symbol to convey the sequence. For example, **Actions** => **Create Test** instructs you to choose the **Create Test** command from the **Actions** menu.

### Syntax diagrams

The following figure shows the standard format for syntax diagrams:
The following example illustrates the syntax for a hypothetical DELETE statement. Because the FROM keyword, alias variable, and WHERE clause are optional, they appear below the main command line. In contrast, the tableName variable appears on the command line because the table name is required. If the statement includes a WHERE clause, the clause must contain a search condition or a CURRENT OF clause. (The searchCondition variable appears on the main line for the WHERE clause, indicating that this choice is required.)

The following guidelines provide additional information about syntax diagrams:

- Read diagrams from left to right and from top to bottom.
- A recursive (left-pointing) arrow above a stack indicates that you may choose more than one item in the stack.
- An underlined item is a default option.
- If a diagram shows punctuation marks, parentheses, or similar symbols, you must enter them as part of the syntax.
- In general, IBM commands, keywords, clauses, and data types are displayed in uppercase letters. However, if an item can be shortened, the minimum required
portion might be shown in uppercase letters, with the remainder in lowercase (for example, CANcel).

- The following conventions apply to variables in syntax diagrams:
  - Variables are typically displayed in lowercase letters and are always italicized.
  - If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words (for example, databaseName).

**Summary of changes**

This topic summarizes product changes and enhancements by version number and release date.

**Version 3.0.00 of MAXM Database Advisor for IMS, and version 5.0.00 of Backup and Recovery Solution for IMS, December 2015**

This release fixes known problems and includes the following enhancements:

- This release includes changes to the BMC Database Management Console, including the following items:
  - Perspectives display data for features in a way best suited for the feature
  - The menu bar no longer appears at the top of the console. Instead, to access console-wide functions (including the online Help), click your user name in the top-right corner of the console.
  - Support for version 1.0 of the Transport Layer Security (TLS) protocol
  - The Schedules window provides a Run button. You can click this button to submit a selected schedule for immediate execution.

- You can use the new scheduled function, REXX EXEC SCHEDULE, to queue a REXX EXEC to execute at a scheduled day and time when an exception is detected. You can configure REXX EXEC SCHEDULE to control when REXX EXEC batch jobs are generated for execution.

The following new REXX EXEC parameters support the REXX EXEC SCHEDULE function:

- Invoke REXX EXEC
- REXX EXEC Name
— REXX EXEC Library
— Defer REXX EXEC until scheduled

■ The new Utility Migration JCL Generator creates the JCL to run the basic functions of BMC database products for IMS. You can use the JCL Generator to get going quickly with these products.

■ The new IMS Status Monitoring feature detects and reports problems in IBM IMS online systems. New parameters support this feature.

■ The new Threshold Settings batch report lists threshold settings, according to filters you select. You use the DLIATRPT utility to request the report. The utility writes the report to a data set, from which you can view and analyze the data.

■ New reorganization threshold parameters are available, including:
  — For full-function databases, high availability large databases (HALDBs), and BMC Partition Database Facility (PDF) databases:
    — CI Splits Count
    — CA Splits Count
    — Required CI Size
  — For Fast Path databases:
    — Area Used %
    — Reclaimable IOVF for UOW %
    — IOVF UOW's less than total UOW's %
    — DOVF CI's less than CI's per UOW %
    — Required CI Size

■ New Recovery Advisor parameters are available, including:
  — Recovery needed count is invalid
  — PRILOG Record Size %
  — RECON Warning Days
  — RECON Detail Record Retention (days)

■ A new HTML Format parameter determines whether to use HTML format for e-mail notifications.

■ Database Advisor reports new exceptions, including:

<table>
<thead>
<tr>
<th>Exception name</th>
<th>When reported</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIDAM PRIMARY INDEX ERROR</td>
<td>Data collection hash checking detects that the hash value for the HIDAM primary index pointers does not match the hash value for the HIDAM primary database root-segment, relative-byte addresses.</td>
</tr>
</tbody>
</table>
Exception name | When reported
---|---
POINTER ERROR | Data collection hash checking detects a pointer error in the database.
FSE ERROR | Data collection detects a free space element error in the database.
UNALLOCATED DATASET | Data collection detects that a required database data set is not allocated.
UNINITIALIZED DATASET | Data collection detects an uninitialized database data set.
RECLAIMABLE SPACE EXCEEDED | The number of bytes occupied by deleted segments that can be reclaimed by reorganizing the database has exceeded the value specified.
RECLAIMABLE FREESPACE EXCEEDED | The percentage of space occupied by deleted segments that can be reclaimed by reorganizing the database has exceeded the value specified.
TOO FEW EMPTY RAPS | The percentage of empty RAPs in the RAA meets or falls below the value specified.
MAXIMUM VSAM ALLOC EXCEEDED | The percentage of allocated cylinders for a VSAM database data set has exceeded the value specified.
MAXIMUM OSAM ALLOC EXCEEDED | The percentage of allocated cylinders for an OSAM database data set has exceeded the value specified.
RECOVERY NEEDED COUNT INVALID | A DBD, DEDB, or HALDB partition has a recovery needed count that does not match the number of recovery needed flags turned on for its DSGs or areas.
PRILOG SIZE EXCEEDS THRESHOLD | An open online PRILOG record is approaching or has exceeded the specified percentage of the maximum 16-MB record size.

- This release supports the new Catalog Directories feature in Version 14.1 of the IBM IMS system. IMS Catalog Directories can be used in place of or in conjunction with the DBDLIBs and ACBLIBs specified.

- This release supports the HALDB OSAM 8 Gigabyte option and the OSAM DEB 24 bit storage VSCR. For example, the ESTIMATE command now supports the OSAM8GB keyword, which determines whether Advisor models 8 GB OSAM partition data sets.

**Version 2.5.00 of MAXM Database Advisor for IMS, and version 4.6.00 of Backup and Recovery Solution for IMS, January 2014**

This release fixes known problems and includes the following enhancements:

- The new DBRC List History window graphically represents the information that DBRC provides when you enter the LIST.HISTORY command. You can view the window for IMS full-function databases, HALDB partitions, and Fast Path data entry database (DEDB) areas. From this window, you can also initiate DBRC commands for the selected object, and monitor the results.
The new Historical Analysis Wizard help you analyze Advisor historical data for statistics of interest. These statistics are usually database statistics that have a rate of change above a certain threshold.

The new Database Exception batch report lists existing exceptions of interest, based on filters that you select. The report provides the same information as the Exception List window in the Database Management Console. You use the EXCEPLST command of the CPCBATCH utility to request the report. CPCBATCH writes the report to a data set, from which you can view and analyze the data.

You can use the new SUMRYRPT command of the CPCBATCH utility to request any of the following batch reports:

- Database History
- Dataset History
- Dataset Space
- Area Performance
- Area Space
- Database Summary

CPCBATCH writes the report to a data set, from which you can view and analyze the data.

Recovery Advisor thresholds, exceptions, and solutions are now available when you have a license for the MAXM Database Advisor for IMS product.

The former JCL Generation Status window and repository are now called the Workflow Status window and repository. The window displays statuses from background processes, such as the Recovery Wizard and the Historical Analysis Wizard. You can now access this window from the right-click menu for a RECON (IMSplex) object.

The Recovery Wizard now supports recovery to alternate database data sets.

The Filter dialog now provides enhanced filter options for the Exception List window.

Database Advisor provides several new threshold parameters:

- Total Freespace(LS) % and Alloc Freespace(LS) % monitor the amount of free space that can accommodate the largest segment size. When these thresholds are exceeded, the product reports the INSUFF TOTAL FREE SPACE (LS) and INSUFF ALLOC FREE SPACE (LS) exceptions to the Exception List. You can view related statistics in the Data Set State and Data Set History reports.

- RAA Space Utilization % and DOVF Space Utilization % monitor space usage in root addressable area and dependent overflow area for Fast Path DEDB areas. When these thresholds are exceeded, the product reports the
AREA RAA UTILIZATION TOO HIGH and AREA DOVF UTILIZATION TOO HIGH exceptions to the Exception List. You can view related statistics in the area reports.

- Database Advisor now supports partitioned databases that are created and maintained by using the BMC Partitioned Database Facility (PDF) product. The interface displays PDF objects in the Navigation window. Labels and fields in the interface indicate that PDF is a supported database type.

- A job to execute the FF COLLECT or FP COLLECT function now processes a maximum of 240 databases or areas. If more than 240 require processing, Advisor initiates multiple COLLECT jobs, as needed. To ensure that these jobs have unique names and can run concurrently, you must add the GROUP SCHDTASK(M) keyword to your CPCxINI procedure. For more information, see the *Database Products for IMS Customization Guide*.

**Version 2.4.00 of MAXM Database Advisor for IMS, and version 4.5.00 of Backup and Recovery Solution for IMS, December 2011**

- The Recovery Wizard creates JCL for executing the BMC Recovery utility. Using this step-by-step process, you can recover all objects in a database group quickly and easily. You can view the status of Recovery Wizard requests in the JCL Generation Status window, and view, edit, and submit the JCL for execution.

- To create recovery JCL, this release adds the following parameters for JCL generation:
  - Number of RVP IC Copies
  - Use RVP Automatic Restart
  - Checkpoint data set model
  - CA Extract model
  - Log Extract model
  - Rebuild Indexes—Primary and Secondary Indexes
  - Rebuild Indexes—Indirect List Data Sets (ILDS)

- The High Availability Large Database (HALDB) modeling and migration tools provide a new workflow approach to managing HALDB-related tasks. While gathering interactive information from you during the workflow process, Advisor creates JCL, control statements, and definitions in the background. You can then access and work with the results in the Maintain Existing Workflow window.

  The following HALDB modeling and migration tools are available:
  - HALDB Migration Workflow Definition wizard
  - Generate Model wizard
  - Partition selection exit (PSE) Validation wizard
  - Rebalance HALDB wizard
— Split Partition wizard

- Recovery Advisor can report the PREOPEN NOT SET FOR AREA exception when a Fast Path area does not have the PREOPEN option set in the RECON. The Area is not set to preopen parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS PREOPEN solution to correct this exception.

- The Solution utility now supports the percent sign (%) in masked values. A mask can contain one or more percent signs (%) and asterisks (*). Use the percent sign to mask a single character. Use the asterisk to mask multiple characters.

- History Compare reports display historic information about multiple databases and partitions so that you can visually compare the information in a graph or table format. You can view database and partition statistics and area space and area performance statistics for comparison.

- Advisor can now report Job History statistics for stand-alone BMC Image Copy, Recovery, and Change Accumulation utility jobs. The utilities send these statistics to Advisor if you have specified the JOBHIST(Y) option in the global options or in the control statement data set for the job.

- The existing Reorg Thresholds Warning Days parameter specifies a default value to use if analyzing the RECON data sets or if one of the following new database Warning Days by Threshold Type parameters does not apply:
  - Warning Days - Data Set Size
  - Warning Days - Extents & Volumes
  - Warning Days - Free Space
  - Warning Days - Database Performance
  - Warning Days - CA & CI Splits

- You can now submit a batch request to perform an analysis for all database objects in a RECON (IMSPLEX). In the Navigation window, right-click the RECON object and choose Request Analysis. (Analysis of RECON data sets is performed during a separate Recovery Collect process.)

- An alternative technique for initiating Advisor scheduled tasks is to use a batch job to execute program CPCTSCHD. This program initiates a specified task immediately when the program is executed, in the same way that the Advisor scheduling system would initiate the task. You can use this program to initiate any task instead of or in addition to initiating that task through the Advisor scheduling system. The execution of program CPCTSCHD does not affect established schedules in the Advisor scheduling system.

- You can perform a periodic audit of your databases to verify that data has been collected and analyzed as expected. The audit verifies all databases that have a defined collection schedule, with the parameter that causes the capture of statistics set to Yes, and with the parameter that defines the collection interval set
to a nonzero value. If the analysis is overdue by five or more days, the audit reports a potential problem as the Not Analyzed (orange) status exception in the Exception List window or as an entry in a batch report.

You can request an audit by using the scheduled AUDIT function (through the console or through the batch CPCTSCHD program) or by using the stand-alone DLIAAUDR program.

**Version 2.3.00 of MAXM Database Advisor for IMS** and **Version 4.4.02 of Backup and Recovery Solution for IMS, November 2010**

- If you have a license for MAXM Database Advisor for IMS, you can use the HALDB Modeling tool in the BMC Database Management Console. This tool helps you migrate full-function IMS databases to HALDBs.

- You can now create and manage dynamic allocation models and matrixes through the Database Management console. They provide a simple, flexible method for managing output data sets (such as image copy, change accumulation, extract, and checkpoint data sets) for backup, change accumulation, and recovery processes.

- Enhancements to the Advisor component's e-mail notification feature are as follows:
  - For the **From** parameter in the E-mail / WTO parameters section of the Maintain Parameters window, you can now provide a parenthetical text string after the e-mail address. For recipients, this string can further clarify the source of the message.
  - The Monitor Recovery function can now generate a notification message when a database recovery fails. This notification message helps DBAs and others to respond quickly to problems, potentially reducing the length of database outages.

- Database Advisor now collects, stores, and reports history information for jobs and job steps instead of for databases and products. You can now access the Job History window at the host-connection level and the IMSPLEX level. (Access is no longer provided at the DBD and partition level.) You can use new options to limit the number of detail records to retrieve and display.

- Recovery Advisor now provides the **RECOVER TO CURRENT BY CAGRP** solution. You can use this solution to generate JCL for recovery of a change accumulation group (CAGRP) to the current state.

- IBM provides the Online Reorganization (OLR) feature for reorganizing a single partition of a HALDB while the other partitions remain available for updates. The OLR feature works through special alternate data sets, which are identified by the suffixes M through V and Y. Advisor now recognizes these data sets when they
are active, considers the associated database to be registered, and allows activities that require database registration to be performed.

- Advisor now provides a solution that you can use to generate JCL for expanding a repository data group. You can access the solution by right-clicking an IMSplex object and select Generate JCL. In the Solutions tab of the Generate JCL window, select the REPOSITORY MAINTENANCE EXPAND DATAGROUP solution.

- When specifying database group criteria, you can now place an asterisk (as a wildcard character) anywhere within the DBD name; previously, you could use an asterisk only at the end of the name. For example, you can specify *CC to select all DBD names that end with the characters CC. With this enhancement, you can specify more complex name patterns than previously supported.
Getting to know Advisor and DBA Toolkit functions

This chapter contains introductory information about the functions and features of BMC database administration products for IMS in the console.

Information requirements

Before you read this chapter, you should have basic knowledge about IMS databases.

You can obtain this knowledge from the IMS documentation set from IBM.

Note

For simplicity, this book uses the term Advisor instead of Database Advisor and Recovery Advisor when a distinction between Database Advisor and Recovery Advisor is unnecessary for clarity or correctness. The book uses the term Database Advisor or Recovery Advisor as needed when the distinction is important.

Products that include console-related functions and features

The following BMC database products for IMS include console-related functions and features.

For more information, see “Functions and features” on page 56.
<table>
<thead>
<tr>
<th>Solution or product</th>
<th>Included Advisor and Toolkit functions and features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup and Recovery Solution for IMS</td>
<td>■ Recovery Advisor functions&lt;br&gt;■ RECON functions&lt;br&gt;■ General functions and features for Advisor support&lt;br&gt;■ Conditional Image Copy feature&lt;br&gt;■ IC Triggering by CA feature&lt;br&gt;■ Monitor Recovery function&lt;br&gt;■ Recovery Wizard&lt;br&gt;■ DBRC List History function&lt;br&gt;■ DBA Toolkit functions</td>
</tr>
<tr>
<td>Fast Path Enhanced Online Suite</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>Fast Path Offline Suite</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>Fast Path Online Suite</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Database Advisor for IMS</td>
<td>■ Database Advisor functions&lt;br&gt;■ Limited subset of Recovery Advisor functions&lt;br&gt;■ RECON functions&lt;br&gt;■ General functions and features for Advisor support&lt;br&gt;■ Conditional Reorg feature&lt;br&gt;■ DBRC List History function&lt;br&gt;■ Historical Analysis Wizard&lt;br&gt;■ DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg for IMS</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg for IMS with Online/Defrag Feature</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg/EP Express for IMS</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg/EP for IMS</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg/EP for IMS with Online/Defrag Feature</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>MAXM Reorg/Online for IMS</td>
<td>DBA Toolkit functions</td>
</tr>
<tr>
<td>POINTER CHECKER PLUS</td>
<td>DBA Toolkit functions</td>
</tr>
</tbody>
</table>

**Summary of concepts**

This section summarizes the main concepts of this chapter.
Data management

As a mission-critical activity, data management focuses on preserving data integrity, improving data availability, increasing data performance, and conserving data management resources.

Data management challenges are greater than ever as:

- The availability of IMS expertise declines
- IMS environments become more complex
- The volume of data to be managed increases
- Demands for continuous availability, greater performance, and cost-savings expand

To meet your data management goals, you must develop a strategy for maintaining the physical and logical structure of your databases, which guards the availability and performance of the data.

“Data management” on page 39 discusses the goals, challenges, and strategies of data management in detail.

Traditional database maintenance cycle

A traditional database maintenance cycle includes the following activities:

- Plan the actions to take (such as reorganizing databases, taking image copies, and checking pointers) and the tools to use.
- Gather information about your environment and databases.
- Analyze this information to detect problems.
- Execute actions to correct the detected problems.

“Traditional database maintenance cycle” on page 41 describes the traditional cycle in detail.

Advisor database maintenance cycle

Advisor directs and automates the database maintenance cycle through processes that reflect and refine the traditional cycle:
In the Configure process, you define your IMS environment to Advisor, typically through an automatic discovery wizard. The wizard guides you through defining IMS systems to Advisor and populating the groups, databases, ddnames, and data sets for each IMS subsystem.

In the Gather process, Database Advisor collects, stores, and manages statistics that are key indicators of database performance and space usage. Recovery Advisor collects information about current conditions that are related to recovery assets and Database Recovery Control (DBRC).

In the Analyze process, Advisor detects and predicts problems by comparing current statistics and conditions to ideal values and conditions. Advisor reports problems as exceptions in the Exception List in a graphical user interface (GUI).

In the Execute process, you address the exceptions in the Exception List by choosing and executing a recommended solution or another available solution.

“Advisor database maintenance cycle” on page 44 describes the Advisor cycle in detail.

Workflow with Advisor

Before you arrive for your workday, Advisor can collect and analyze data about your databases and environment and have the analysis ready in the console. The console is the graphical user interface that provides a central, convenient location for performing database management activities.

In the console, the Exception List specifies the database maintenance tasks that you need to perform, such as reorganizing a database or preventing a database from running out of space. Each entry in the Exception List represents an object that Advisor has flagged with a warning, critical, or dead status.

From the Exception List, you can perform the following tasks:

- View exception details
- List solutions that Advisor recommends for addressing exceptions
- Generate JCL to execute solutions
- View the statistics and parameter values that Advisor uses to detect exceptions

Although you will probably work with Advisor exceptions daily, Advisor also offers other functions and features that you can use as needed.

“Advisor workflow” on page 47 describes the typical daily use of Advisor in detail.
## Functions and features

The following table compares the console-related features and functions that are included in BMC database products for IMS.

### Table 1: Comparison of products that include console-related functions and features

<table>
<thead>
<tr>
<th>Feature or function</th>
<th>MAXM Database Advisor</th>
<th>Backup and Recovery Solution</th>
<th>Other products</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Advisor—Reorg thresholds, exceptions, and solutions</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Recovery Advisor—Recovery thresholds, exceptions, and solutions</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RECON functions:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ RECON thresholds, exceptions, and solutions</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>■ RECON Reorg utility</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ DBRC List History function</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>General functions and features for Advisor support:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Automated schedules for collection and analysis</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>■ On-demand data collection and analysis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Exception List and Solution Manager</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Automated notification of exceptions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Batch utilities to automate JCL generation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC Triggering by CA feature</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Conditional Image Copy feature</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Monitor Recovery function</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Recovery Wizard</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Conditional Reorg feature</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Feature or function</td>
<td>MAXM Database Advisor</td>
<td>Backup and Recovery Solution</td>
<td>Other products</td>
</tr>
<tr>
<td>---------------------------------------------------------</td>
<td>-----------------------</td>
<td>------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Database Advisor tools and reports:</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>■ Historical Analysis Wizard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ HALDB modeling and migration tools</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Disassemble MFS tool</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Partitioning and randomizer tool (batch only)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Exception Forecast report</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Batch reports from the console</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DBA Toolkit functions:</th>
<th>Yes</th>
<th>Yes</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Generate JCL function</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ JES viewer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Disassemble DBD, PSB, ACB, and MDA tools</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ DBD Map tool</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ DEDB space calculation tool</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Build IDCAMS control statements tool</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Object annotation function</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Auto Configure wizard (for discovery of IMS components)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ History reports (creation and maintenance of statistics over time)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ History comparison reports</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Current state reports</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Trend reports</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ Access control (security)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For a list of other products, see “Products that include console-related functions and features” on page 33.

**Summary of benefits**

Advisor revolutionizes your approach to database maintenance tasks.

Automated data collection and data analysis free you from the traditional, manual drudgery of gathering statistics and performing calculations to detect problems. Recommended best-practice solutions and generation of optimized JCL greatly
reduce the amount of time, effort, and skill that are needed to resolve database
maintenance problems. If you must research a problem, Database Advisor keeps the
supporting statistics organized, immediately accessible, and readable through the
graphical user interface.

By handling databases in groups rather than individually, Advisor simplifies
database maintenance tasks. Database Advisor forecasts problems to give you a
chance to address them proactively. When a problem occurs, Advisor notifies you
immediately so that you can correct the problem in a timely way. You can also
secure the product functions to prevent unauthorized access. Advisor works with
and takes advantage of other BMC products in your environment to optimize
resource usage and to provide synergistic benefits from using the products together.

Advisor eliminates the challenges that are typically inherent in implementing new
software and maintaining that software over time. Automated discovery of
components in your IMS environment and sensible default values for processing
parameters reduce setup time. Consequently, Advisor can start performing
productive work almost immediately after installation. Self-maintaining data
repositories almost eliminate product administration activities, and outstanding
BMC customer support is always available to help you with questions and problems.

“Benefits” on page 65 explains how Advisor is powerful; flexible; and easy to use,
implement, and maintain.

Data management

Data management is a mission-critical activity in almost every IT organization today.
Data managers work to reach critical goals, conquer major challenges, and employ
strategies that ensure success.

Goals

Your organization strives to achieve the following critical goals:

- Preserve data integrity

  System failures, application and procedural errors, and disasters can threaten the
  accuracy and completeness of your valuable data. You must design and
  implement processes that safeguard data integrity.

- Improve data availability

  You must meet the needs of customers and employees for access to the data while
  handling planned and unplanned outages. Service-level agreements specify
  availability requirements that dictate maximum allowable downtime. You must
design and implement processes that help you get data back online swiftly after an outage.

- Increase data performance
  You depend on IMS to process vast amounts of data with unparalleled speed and efficiency. To maintain optimal data performance, you must design and implement processes to keep your IMS databases organized and well defined.

- Conserve data management resources
  Every organization has limited resources—time, money, software, hardware, and skilled people—to handle routine data management tasks, special projects, and occasional problems. You must implement processes that spend precious resources wisely, with maximum payback and minimum waste.

**Challenges**

Several major challenges make the goals of data management difficult to accomplish:

- Diminishing availability of IMS expertise
  The problem of replacing IMS professionals who retire or leave the IT field is likely to grow over time. You must find a way to capture the expertise of knowledgeable IMS DBAs so that you can continue to meet your data management goals when these DBAs are no longer available.

- Complexity of the environment
  Enterprise environments are becoming increasingly complex. Many sites support multiple z/OS images, databases that are shared among multiple IMS systems, and applications that span multiple types of database management systems. You must find ways to simplify your everyday tasks, even if you cannot simplify your environment.

- Growing volume of data
  The amount of data in IMS databases worldwide is growing. As the volume increases, managing that data becomes increasingly difficult and costly. You must find a way to manage the expanding volume of data, even if the resources for managing this data remain flat or are reduced.

- Expanding demands
  IT departments face ever-increasing demands to process more data faster, more efficiently, and with fewer resources. Key databases must be available 24 hours a day, 365 days a year. Batch applications must complete within a shrinking batch window. Users expect immediate responses from interactive applications. When budgets are cut, you are told to do more with less. You must find a way to meet expanding demands.
Strategies

A database management system such as IMS provides structures, processes, services, and tools for handling data safely, quickly, and efficiently.

However, the general efficiency and reliability of IMS alone are insufficient. To reach data management goals and conquer data management challenges, you must develop dependable database management strategies. These strategies include one for maintaining the physical and logical structures of your databases.

Your database maintenance strategy affects the performance of your databases and the productivity of DBAs. A maintenance strategy ensures that problems are identified and corrected before degrading database performance. Your strategy should prescribe a routine for keeping databases organized, online, and running smoothly.

Your operating system, IMS, and BMC products work together to support a robust maintenance strategy.

Traditional database maintenance cycle

The traditional database maintenance cycle has been the same since IMS was introduced.
Figure 1 on page 42 shows the general processes in this cycle.

Figure 1: Traditional database maintenance cycle

You can refine this cycle by employing tools and techniques to improve the reliability, usability, and performance of the these processes and their related tasks, while reducing resource usage. However, when the related tasks are performed with traditional tools and techniques, they are unavoidably time-consuming, difficult, and burdensome. As explained in “Advisor database maintenance cycle” on page 35, Advisor automates the traditional database maintenance cycle to reduce the time, effort, and skill that are required for performance of related tasks.

**Traditional Plan process**

The traditional Plan process involves researching and making decisions about the tools and techniques that you will use to implement your maintenance strategy. The ideal approach is to determine which tasks and goals are necessary and then to find the tools that are best designed to help you carry out your plan.
You must examine the techniques to use for the Gather, Analyze, and Execute processes of your strategy. Key considerations may include identifying the specific data elements that you must collect about your environment and databases, formulating a method of analysis, developing a monitoring schedule, and designing a system of rules that tell you when and how to react to the analysis.

Traditional Gather process

The traditional Gather process involves collecting and managing information about your environment and databases.

You must build and run the jobs to collect data. You also must manage the storage of the data that is gathered, which includes performing regular backups, purging data according to established retention schedules, and recovering data when necessary.

The Gather process provides all information on which you base your analysis. You must collect key elements about the state of each database. You need to know about any events which occurred in your system that may affect the databases. While you must ensure that the data which is gathered is current, you also need historical details about your databases. Using the historical details, you can identify and chart data trends.

Traditional Analyze process

The traditional Analyze process involves organizing and interpreting the gathered data. You must process the data into meaningful and manageable reports. You must build and run jobs to generate the reports. Finally, you must read, compare, and interpret the reports.

The Analyze process is important because it identifies problems with your databases. Because it would be impractical, if not impossible, to review reports for thousands of databases manually, the Analyze process must be efficient and consistent. You need a method of analysis that pinpoints specific database problems and delivers repeatable results.

Traditional Execute process

The traditional Execute process involves taking action to correct problems that were found during the analysis. You must decide which solutions to execute and then build and run the jobs. You also must monitor the jobs and review job output.
The Execute process is important because the overall health and performance of your databases depend on taking the appropriate action at the appropriate time. Because you must limit the frequency and duration of taking databases offline, the Execute process must use resources economically. You must execute solutions that correct as many problems as possible with as few resources as possible.

Database administration processes are cyclical. When solutions have been executed to correct problems that were reported by the analysis, the cycle begins anew. You must collect current data, analyze the data to identify problems, and then execute the appropriate solutions to correct those problems. To implement your database administration strategy, you may also have to assess and modify the plans that you made.

**Advisor database maintenance cycle**

BMC has refined and automated the basic processes of the general database maintenance cycle to provide a robust database maintenance solution in Advisor. Advisor provides a complete, integrated solution that directs and automates the database maintenance cycle.
Figure 2 on page 45 shows the Advisor database maintenance cycle.

**Figure 2: Advisor database maintenance cycle**

1. **Configure**
   - Advisor recommends solutions to correct the problem and lets you execute the solution
   - You run the auto-configure feature and review suggested default analysis options

2. **Gather**
   - Advisor gathers space and performance information about the state of a database

3. **Analyze**
   - Advisor analyzes a database and reports the current and potential database problems

4. **Execute**
   - The ADVISOR process AUTOMATES database maintenance

Advisor uses state-of-the-art techniques to accomplish the following tasks:

- Gather information about your databases and environment

- Proactively predict problems

- Recommend the best course of action for preventing or correcting detected problems

You interact with Advisor through the console, a convenient Windows-based user interface.

**Advisor Configure process**

The Advisor Configure process replaces the Plan process in your general database maintenance cycle. Instead of spending countless hours and resources planning a
strategy, you can implement Advisor with little preparation, and Advisor starts working immediately.

The Configure process includes identifying your IMS systems and populating the systems with groups and databases. Advisor automates configuration activities through automated discovery techniques.

The Configure process also includes JCL-customizing parameters that are used to control other processes. The default values for most parameters work for most Advisor users, so you probably will not have to make many changes.

Advisor stores configuration information and customized parameter values in self-managing repositories. The functions that you can schedule to back up repository data, purge repository data, and synchronize database registration information virtually eliminate product administration tasks.

**Advisor Gather process**

The Advisor Gather process automates collection, storage, and management of data about your databases and environment. Advisor stores gathered data in self-managing repositories.

During the Gather process, Advisor collects key indicators of database performance and space usage or indicators of current conditions in the environment. This data is evaluated during the Analyze process.

**Advisor Analyze process**

The Analyze process works differently for Database Advisor and Recovery Advisor:

- The Database Advisor Analyze process uses the key database performance and space indicators that Database Advisor collected during the Gather process to detect database problems and to predict them proactively. The Analyze process is initiated to execute automatically after any new data has been collected about your databases.

  The Database Advisor Analyze process uses base values, current state values, and threshold values to identify and predict database problems. Base values reflect the optimal state of a database, which normally follows a database reorganization. Current state values are the most current data that Database Advisor has collected about a database. Threshold values are user-defined, maximum acceptable deviations from base values. A threshold represents a limit that the current state value should not exceed.

  During analysis, Database Advisor uses the base value and the current state value to calculate the rate at which a threshold is being approached and to detect
whether the threshold has been met or exceeded. When a threshold violation occurs or is predicted to occur, Database Advisor lists the database on the Exception List with a description of the problem.

- The Recovery Advisor collection and analysis process, which is known as Recovery Collect, reads the DBRC RECON data sets and analyzes the records to detect conditions that can affect database recovery and RECON status. You identify the conditions of interest in your environment by specifying threshold parameter values. The process runs automatically on a schedule that you establish or on demand.

If the Analyze process finds a problem, Advisor reports the problem (called an exception) in a central location (the Exception List) in the console. The Exception List is an interactive table where you can view the details about an exception, correct all (or selected) exceptions, and research an exception.

**Advisor Execute process**

The Execute process guides you through the process of correcting exceptions that have been reported on the Exception List.

With Advisor, you can take action on all reported exceptions across any level of defined components: sysplex, IMSPLEX (as represented by a defined DBRC RECON), IMS system, group of databases, database, High Availability Large Database (HALDB) or BMC Partitioned Database Facility (PDF) partition, or database data set or area.

Determining the action to take in response to an exception is also simplified. Advisor presents all solutions that are available to you and recommends the best solution for each exception (or combination of exceptions). The recommended solution is based on the utilities that you have installed and on user-defined goals that tell Advisor about your general priorities (the relative importance of performance, availability, and space usage).

Advisor incorporates the Generate JCL function, which you can use to generate and display JCL for a recommended solution. This solution corrects all problems that are reported or forecast for the database. When you review the generated JCL, you can submit the job to correct the problems and remove the database from the Exception List.

**Advisor workflow**

This section demonstrates how you interact with Advisor on a daily basis and how Advisor works in response.
This section suggests how you can perform database administration tasks more quickly, easily, consistently, and accurately with Advisor than with traditional methods.

Figure 3 on page 49 shows the logical structure of Advisor and the flow of action for the major functional areas. For a detailed diagram and explanation, see “Getting started with Advisor and DBA Toolkit functions” on page 71.
Figure 3: Logical structure of Advisor
Graphical user interface

The console is the graphical user interface for Advisor and the DBA Toolkit.

This console is the central location from which you manage your database maintenance tasks. Figure 4 on page 50 shows the console with a typical IMS database administration environment that has been defined.

Figure 4: Console and IMS database administration environment

You launch the console from the Windows Start menu on your desktop. During launch, the console automatically checks for and notifies you about available updates. Installing console maintenance is a user-controlled, one-click operation which is similar to the initial console installation that you initiate from your web browser.

When the console opens, you connect to a host (by right clicking the host object) to access IMS applications and objects (IMS systems, groups, databases, data sets, and so on) on the mainframe. You are prompted to enter your TSO user ID and password.
(which can be saved so that you can connect to a host later by clicking the host object).

In the console, you access information and actions for the IMS components by clicking their displayed icons in the Navigation window (the object tree). The tree represents a hierarchy of IMS components. From the highest level to the lowest level, the following objects are included in the IMS component hierarchy:

The console displays the results of your requests in the work area. You can page among multiple results by clicking buttons along the bottom of the work area. You view product messages in the Messages window.

**Off-hour processing**

Advisor processes typically are set to run when system and database resources are least active.

**Database Advisor Collect process**

The Database Advisor Collect process looks at your databases and your environment.

Database Advisor checks each database to determine whether it is due for data collection. Database Advisor bases its determination on a data collection interval parameter that has been defined for the database. Database Advisor also checks your environment to analyze the events of the day and to determine other actions that might have to be performed. For example, if a reorganization was run during the day, Database Advisor might have to reset the base values for a database. If a data
repository is filling up quickly, Database Advisor might have to increase repository space allocation.

**Database Advisor Analyze process**

When the Database Advisor Collect process runs, it stores the new data about the current state of your databases in the repositories.

When the repositories receive new data about your databases, the Database Advisor Analyze process runs. The Analyze process checks the new current state values against the base values and thresholds to find and forecast problems.

The Database Advisor Analyze process populates the Exception List with an exception record for each threshold violation with the user-specified time range. The record identifies the database that was analyzed, the current status of the threshold, and the dates at which the threshold is expected to reach the following status severities:

**Warning**

The threshold is expected to be met or exceeded, and preventive action is recommended.

**Critical**

The threshold has been met or exceeded, and corrective action is required.

**Dead**

The database has reached an operating system limit, and immediate action is required.

The Analyze process stores exception records in self-managing data repositories.

**Recovery Advisor collection and analysis process**

The Recovery Advisor collection and analysis process (Recovery Collect) examines IMS RECON data sets for conditions that can cause problems with database recovery or operations.

For example, a recovery asset (image copy, change accumulation, or log data set) that is marked in error in the RECON data sets cannot be used in a recovery. Too many logs that have been written since the last image copy or change accumulation might cause the outage for a recovery to be longer than allowed by your service level agreement. A database might be in recovery-needed status. A RECON data set might be unavailable or require reorganization to improve performance.
You can control which conditions that Recovery Advisor detects by setting threshold parameters. If Recovery Advisor detects a problem, it reports an exception to the Exception List.

**Exception list**

After launching the console, you can access the Exception List to view the exceptions that Advisor found during the last analysis process.

**Figure 5: Exception List**

The Exception List corresponds to a list of data management tasks that must be performed, such as reorganizing a database, taking immediate action to prevent a database from running out of space, adjusting the amount of space in the root addressable area for a database, recovering a database, taking an image copy, or researching why a particular exception was reported.

The Exception List contains an entry for each object (database, change accumulation, log, or RECON) that has one or more exceptions with a status of *warning*, *critical*, or *dead*. For Database Advisor and RECON exceptions, the entry also lists the dates at which the exception is projected to reach the next status severity, if applicable.

The Exception List includes a Summary table, which identifies all objects for which projected or actual exceptions have been detected, and a Details table, which lists the individual threshold violations for the object that you select in the Summary table.
You display the Exception List by right-clicking any appropriate object in the Navigation window and selecting **Exception List**. The Exception List contains all exceptions for the selected object. For example, if you display the Exception List for an IMS system, the Exception List contains all exceptions that have been reported for all objects in the IMS system.

### Problem resolution

When an object is listed on the Exception List, you can use the BMC-recommended solution to correct all exceptions that have been reported for that object. Or you can choose specific exceptions that have been reported for the object and use the recommended solution (or any available solution) to correct the problems.

To use recommended solutions, you select an object in the Summary table of the Exception List and click **Solutions**. Figure 6 on page 54 shows the Solutions window, where you can select the exceptions that you want to correct, display all available solutions, generate JCL for a solution, review the generated JCL, and submit the job.

---

**Note**

To use the BMC-recommended solution, you can right-click the exception in the Summary table and choose **Recommended JCL**.

---

**Figure 6: Solutions window**

When the list of available solutions is displayed, the order is based on available utilities and your specified goals that establish space, performance, and availability priorities for the database. For example, if you specify the **Can take offline briefly** goal, Advisor recommends solutions that require only a brief outage.
Statistical research

If you do not immediately recognize or understand the reasons why Advisor reported an exception, you can review the current state values, the base values, and the threshold values that were analyzed to detect the exception.

This review might show that you need to adjust a threshold value or reset the base values for a database. You can also review the history reports that indicate data trends for the database.

Figure 7 on page 55 shows the Data Set State report.

![Data Set State report](image)

Processing options

Advisor includes default values for most parameters that control data collection and analysis, and you might be able to accept the defaults without change. You can modify parameter values to suit your environment or the special requirements of an object, such as a group or a database.
You customize parameters by using the console. Figure 8 on page 56 shows a typical dialog box for editing parameters—in this case, for controlling Database Advisor thresholds for full-function databases and HALDBs.

**Figure 8: Maintain Parameters window**

To simplify the management of parameter values, you can set parameters at various levels in the hierarchy of objects. Objects at lower levels in the hierarchy "inherit" the values that you set at higher levels. When you customize a parameter for an object, all objects that are subordinate to the object inherit the new parameter value unless you also customize the same parameter for the subordinate object. The console displays inherited values and the source of inheritance for all parameters.

**Functions and features**

This section describes the major functions and features of BMC database products for IMS that are available in the console.

**Database Advisor functions**

Database Advisor provides support for Reorg thresholds, exceptions, and solutions.
This support is used to detect and correct common database problems that affect performance and space usage. Scheduled functions automatically collect statistics about your databases, analyze the statistics against threshold parameter values that you have established, and report existing and future problems as exceptions on the Exception List. You view the Exception List in the console and can solve reported problems by selecting solutions, which generate JCL to correct exceptions.

Database Advisor is included with the MAXM Database Advisor product for IMS.

**Recovery Advisor functions**

Recovery Advisor provides support for Recovery thresholds, exceptions, and solutions.

This support is used to detect and correct common problems that affect the recoverability of IMS databases and the performance of RECON data sets. A scheduled function automatically reads the Database Recovery Control (DBRC) RECON data sets, analyzes the records against threshold parameter values that you have established, and reports problems as exceptions on the Exception List. You view the Exception List in the console and can solve reported problems by generating and executing solutions.

Recovery Advisor is included with the Backup and Recovery Solution *for IMS* product and the MAXM Database Advisor *for IMS* product.

**RECON Advisor functions**

RECON Advisor functions help you manage and work with your RECONs:
- RECON thresholds, exceptions, and solutions
- RECON Reorg utility
- DBRC List History function

You can use RECON thresholds, exceptions, and solutions to detect and correct common problems that affect the availability and performance of RECON data sets. A scheduled function automatically reads the DBRC RECON data sets, analyzes current values for the RECON data sets against threshold parameter values that you have established, and reports problems as exceptions on the Exception List. You view the Exception List in the console and can solve reported problems by generating and executing solutions.

The RECON Reorg utility restores DBRC RECON data sets to optimum availability and performance levels, and resets base values for the statistical information that Advisor collected for the RECON data sets.
The DBRC List History function is a GUI representation of the data the DBRC returns from the LIST.HISTORY command. You can view the data in a table or a timeline format. In the GUI, you can click certain events and tabs to submit DBRC commands or generate JCL. For example, you can submit a command to change the GENMAX value for a database data set, or generate JCL to create an image copy or recover a database to a point in time.

RECON Advisor functions are included with Backup and Recovery Solution for IMS and MAXM Database Advisor for IMS.

**General functions and features for Advisor support**

The MAXM Database Advisor for IMS product and the Backup and Recovery Solution for IMS product provide functions and features to support Database Advisor, Recovery Advisor, and RECON functions.

**Automated schedules**

Advisor provides schedules that you can use to automate Advisor activities such as Database Advisor data collection and analysis processing, Recovery Advisor Recovery Collect processing, e-mail consolidation, and repository management processes.

You can customize and manage schedules from the console.

**On-demand data collection and analysis**

At any time, you can request data collection and analysis from the console by right-clicking an object (such as a database or RECON) and choosing a command from the right-click menu.

**Exception List and Solution Manager**

Advisor provides a central Exception List that provides a comprehensive view of all problems (and potential problems) in your IMS environment. Through this list, DBAs can select solutions to correct the problems. The Advisor Solution Manager recommends solutions to correct all problems or individual problems for a selected object.

The Database Exception batch report provides the same information as the Exception list provides. You request the report by submitting a job to execute the CPCBATCH utility. The utility writes the report to a member of a partitioned data set.
Automated notification

Advisor can send e-mail messages, WTO messages, or both to notify specified users of detected exceptions. Advisor can consolidate messages into a single message and handle them differently, based on their severity.

Batch utilities to automate JCL generation

You can automate routine tasks by scheduling the execution of Advisor batch utilities. When an Advisor exception indicates that a routine task should be performed, these utilities detect the exception, generate the JCL to correct the exception, and submit the JCL for execution.

IC Triggering by CA feature

You can use the IC Triggering by CA feature to balance change accumulation (CA) and image copy (IC) processing for one or more CA groups.

The feature manages the size of the CA data set for a CA group through automated IC processing of the objects in the group. This automation ensures that the CA data set does not grow too large to be processed effectively during database recovery.

The IC Triggering by CA feature works with the BMC Change Accumulation utility and the BMC Image Copy utility. To use the feature, you must have a license for the Backup and Recovery Solution for IMS product.

Conditional Image Copy feature

You can use the Conditional Image Copy feature to bypass a scheduled stand-alone batch image copy task for an object (a database, partition, or area).

The task is bypassed if the object has not been updated since the last image copy was performed and if an image copy is not required. The purpose of the feature is to avoid taking unnecessary and extraneous image copies, saving processing and data storage resources with no compromise to data integrity.

The Conditional Image Copy feature works with the stand-alone batch image copy function of the BMC Image Copy utility and requires a license for the Backup and Recovery Solution for IMS product.
**Conditional Reorg feature**

The Conditional Reorg feature permits a submitted reorganization job step for a database to proceed only if Database Advisor has detected or forecast an exception that can be solved by database reorganization.

If the database is not flagged as requiring reorganization (because an exception has not occurred or is not forecast), the reorganization job step is terminated.

The Conditional Reorg feature is included with MAXM Database Advisor for IMS. The feature works automatically with BMC products that are based on MAXM Extended Performance (MAXM/EP) architecture. The feature works as a stand-alone utility with BMC products that are based on "Classic" architecture. The Conditional Reorg feature applies to full-function databases and high availability large databases (HALDBs) only; it does not apply to Fast Path areas.

**Monitor Recovery function**

The Monitor Recovery function receives information about recovery processing from the BMC Recovery utility and displays this information in the Monitor Recovery window and dependent windows in the console.

You can use the Monitor Recovery function to watch the progress of recovery job steps, to determine whether a problem has occurred, and to check on the recovery of objects.

The Monitor Recovery function is included with Backup and Recovery Solution for IMS.

**Recovery Wizard**

The Recovery Wizard creates JCL for executing the BMC Recovery utility. Using this step-by-step process, you can recover all objects in a database group quickly and easily.

You choose a group to recover, the recovery point, and whether to create JCL that simulates a recovery or performs an actual recovery. In addition, you can choose to recover to alternate data sets. You can accept established values for recovery parameters or specify override values. You must also specify an output data set and member to contain the generated JCL.

Using your choices and specifications, the wizard generates the JCL in the background, and notifies you when the generation is complete. You can then view
the status of the request in the Workflow Status window, and view, edit, and submit the JCL for execution.

To use the Recovery Wizard, you must have a license for Backup and Recovery Solution for IMS.

MAXM Database Advisor tools and reports

In addition to the tools and reports that are available with the DBA Toolkit, MAXM Database Advisor exclusively offers the following tools and reports:

Historical Analysis Wizard

The Historical Analysis Wizard in the BMC Database Management Console guides you through the process of analyzing Advisor historical data for statistics of interest. These statistics are usually database statistics that have a rate of change above a certain threshold. You use the wizard to select parameters for the analysis. The wizard runs the analysis in the background and notifies you when the analysis is complete. You can then view the results of the analysis in the Workflow Status window.

HALDB modeling and migration tools

The HALDB modeling and migration tools in the BMC Database Management Console provide information and recommendations that can help you migrate traditional full-function IMS databases to high availability large databases (HALDBs) and manage existing HALDB partitions.

The migration tools extract information from an existing database or secondary index. Then, using the extracted information and your proposed specifications, the tool recommends values to define the following items, depending on your database type:

- Randomizing parameters and key ranges for a partitioned HDAM (PHDAM) database
- Key ranges for a partitioned HIDAM (PHIDAM) database or partitioned INDEX (PSINDEX) database

When you are satisfied with the recommendations, the tool generates sample JCL, control statements, and definitions to set up the new HALDB.

Other tools help you rebalance data between partitions and split the data in a partition into multiple partitions.
The HALDB modeling and migration tools provide similar results as the batch partition and randomizer tuning tool, but in an interactive format that simplifies the process and improves DBA productivity.

**Disassemble MFS tool**

The message format service (MFS) control block disassembly tool offers a quick and convenient way to produce source code for MFS control blocks.

**Partition and randomizer tuning tool**

The partition and randomizer tuning tool is a batch utility that selects hierarchical indexed direct access method (HIDAM) key ranges or hierarchical direct access method (HDAM) parameters to convert a full-function database into a high availability large database (HALDB).

The tool recommends partition boundaries. You can use this tool to obtain randomizing parameters for HDAM databases. You can also use this tool to determine the optimal ranges for your record keys so that data is redistributed evenly across partitions.

**Exception report**

Database Advisor provides the Exception Forecast report for projecting future exceptions that are based on the rate at which a database is approaching a threshold. The Exception Forecast report can be generated for specific date ranges and exception status types.

**Batch reports from the console**

Database Advisor can generate batch reports from the IMS Database Management console. These reports provide a high-level, consolidated view of all databases in your environment. You can visually compare data for multiple databases and export the data for analysis by a third-party application.

**DBA Toolkit functions and tools**

The DBA Toolkit includes the following functions and tools.

The DBA Toolkit is available with all products that are listed in “Products that include console-related functions and features” on page 33".
Generate JCL function

The Generate JCL function makes it easy to generate, edit, submit, and store JCL that is optimized for your environment. The JCL generation process is dynamic. When a command to generate JCL is issued, the function evaluates the current state of a database to determine optimal JCL parameters.

The Generate JCL function works with BMC-provided solutions and your own solutions. A solution consists of skeleton JCL to execute a function or utility. The Generate JCL function combines the solution with parameter values, which define customized information and optional processing, to create a customized job that executes the utility or function.

JES viewer

The JES viewer provides a system-wide view of the JES log through the console. You can use the JES viewer to look at the status of current jobs and to review the output for jobs that have been completed.

The JES viewer is not limited to IMS jobs or jobs that are submitted through the console; you can view information about all jobs that write output to the JES log if you have authorization to view those jobs. So that you do not have to sort through potentially thousands of jobs, you can filter the displayed jobs.

Disassemble DBD, PSB, ACB, and MDA tools

The IMS control block disassembly tools offer a quick and convenient way to produce source code for IMS control blocks. The tools disassemble database definition (DBD), program specification block (PSB), application control block (ACB), and dynamic allocation (MDA) control blocks to produce source code for DBD, PSB, ACB, and MDA libraries. The disassembly tools helps you perform the following tasks:

- Rebuild lost source libraries
- Verify the active source version
- Save, change, or reuse source code

DBD map tool

The DBD map tool, as an extension of the disassembly tool, provides an easy way to view, print, and export information about the segment structure of a database, including segment hierarchy, internal logical relationships, and secondary properties.
**DEDB space calculation tool**

The DEDB space calculation tool offers a quick and convenient way to calculate disk space requirements for a Fast Path data entry database (DEDB) area before allocating space.

**Build IDCAMS control statements tool**

The Build IDCAMS control statements tool offers a quick and convenient way to produce and store IDCAMS DELETE, ALLOCATE, and DEFINE CLUSTER control statements for existing data sets.

The generated statements can be used by other utilities that require data sets to be redefined during processing.

**Object annotation function**

You can use the object annotation function to enter free-form information about an object in the console.

For example, you can enter the object’s nickname, unusual characteristics and uses, and reminders about actions related to the object.

**History, Compare History, current state, and trend reports**

Report functions provide a graphical, organized view of the current state and historical statistics for registered objects in the console. This information is collected by certain BMC functions and utilities and is saved to the repositories.

**Auto Configure wizard**

The Auto Configure wizard automates the task of defining your IMS environment and resources to Advisor.

**Access control**

Access control functions secure Advisor and toolkit resources to prevent their use by unauthorized users and processes.
Benefits

This section describes how the functions and features of Database Advisor, Recovery Advisor, and the DBA Toolkit work to automate, streamline, and simplify your data management tasks.

Freedom through automation

The automation in Advisor frees DBAs from the manual work of collecting data about their databases and examining piles of reports to pinpoint problems.

Automatic data collection and analysis make database maintenance activities efficient and effective. Data is collected and analyzed only when necessary and, because parameters such as thresholds and goals are in place, analysis and subsequent recommended solutions are applied consistently.

Thorough analysis of all defined objects and display of exceptions in a central location make it unlikely that problems will be overlooked.

Effortless best solutions

To address identified exceptions, you can trust Advisor to select and generate the best possible solutions.

The BMC recommended solution is easy to access from the Exception List. Each solution is tuned for optimal use of resources, and solutions are suggested in order of their likelihood of correcting the problem. You spend less time and effort working on problems.

Advisor helps less-experienced DBAs learn about (and gain skills to solve) IMS database problems as these DBAs examine reported exceptions and review recommended solutions. Even DBAs who are relatively new to IMS database management can be confident that they are taking the right actions.

Consistent results

Advisor helps to ensure a consistent approach to solving database administration problems.

Because most problems can be solved in a variety of ways, a DBA is likely to choose a different solution than another DBA would choose. With Advisor, the solution
does not depend on which DBA is working on the problem. An optimal solution is generated consistently for every problem.

**Accessible statistics**

The DBA Toolkit automatically creates and maintains an organized collection of current and historical statistics with all information that you need for research about an exception.

These statistics are easy to access from the Exception List, making your research activities efficient and effective. The DBA Toolkit eliminates the need to find, examine, and manually manage clumsy listings that might be cryptically named and might be scattered in data sets throughout the system.

**Simplicity through groups**

With BMC database administration products for IMS, you can manage your IMS databases by creating and acting on groups of databases.

Managing groups rather than individual databases simplifies data management processes, ensures consistent activity across related databases, and increases speed and efficiency of processes. If change accumulation groups or database data set groups are already defined in DBRC, the BMC products recognize and use these groups. Also, based on criteria that you specify, you can create Advisor database groups or use BMC Recovery Manager groups that were created within Recovery Manager.

**Proactive problem-solving**

By forecasting when problems are likely to occur, Database Advisor lets you solve problems before they can attack your data integrity, availability, and performance.

This proactive approach results in efficient and effective planning, scheduling, and resource usage. With Database Advisor, database maintenance activities are performed only when necessary and issues of the most immediate concern receive the appropriate attention.
Automatic notification

Advisor provides automatic notification to help ensure that exceptions are reported promptly and appropriately so that the people who need to know about the problem are told in a timely way and non-critical problems do not wake them up in the middle of the night.

You can set up the notification system to send an e-mail message, WTO message, or both when an exception is added to the Exception List or when the status severity of an exception is changed. For example, your senior DBAs can be notified when an exception with a critical status is detected.

Access control

You can restrict access to Advisor features and functions in the console by implementing access controls.

Access controls are based on system authorization facility (SAF) permissions. You build access controls by defining SAF resources and assigning a type of access, such as READ or UPDATE, to those resources for one or more users. You can restrict access based on functional capability, IMS system, or a combination of functional capability and IMS system.

Synergy with other BMC products

Advisor works with (and takes advantage of) other BMC products in your environment to optimize resource usage and to increase the advantages that you receive from the products.

When the following BMC products, functions, and solutions are executed, data is sent automatically to the Advisor repositories, where the data is available for analysis by Database Advisor:

- MAXM Reorg/Online for IMS
- MAXM Reorg/EP for IMS
- MAXM Reorg/EP for IMS with Online/Defrag Feature
- MAXM Reorg/EP Express for IMS
- UNLOAD PLUS/EP for IMS
- LOADPLUS/EP for IMS
- SECONDARY INDEX UTILITY/EP
- FAST REORG FACILITY/EP
- POINTER CHECKER PLUS
BMC Fast Path suites
- Fast Path Analyzer/EP
- Fast Path Online Analyzer/EP

Recovery Advisor works with BMC functions and utilities that are included in the Backup and Recovery Solution for IMS product. Recovery Advisor generates JCL for the Image Copy utility, the Change Accumulation utility, and the Recovery utility. The Monitor Recovery function communicates with the Recovery utility to track database recovery processing.

BMC system administration products for IMS can coexist in the same console with the BMC database administration products for IMS.

**Fewer unnecessary processes**

A traditional database maintenance strategy for full-function databases and HALDBs typically involves scheduling reorganizations and image copies to occur on a routine basis, regardless of whether this processing is necessary. The result is often needless database unavailability and wasted system resources.

With the Conditional Image Copy feature, image copies are performed for a database, partition, or area only when updates have occurred for the object or when an image copy is required. If a scheduled image copy is not needed, it is not performed.

With the Conditional Reorg feature, reorganizations of full-function databases and HALDBs are performed only when indicated (because Database Advisor has identified a problem that can be solved by reorganization) rather than on an arbitrary schedule (because the true status of the database is unknown but a reorganization might be required). The Conditional Reorg feature bypasses reorganization unless the Database Advisor analysis process indicates that a reorganization is necessary.

**Better recovery monitoring**

The Monitor Recovery function is helpful any time that you want to track a recovery. The Monitor Recovery function is especially suited to disaster recovery scenarios when a large number of recoveries are executing simultaneously. With icons to highlight problems, problems are less likely to be overlooked than they are in a traditional interface.

The graphical view of the recovery environment can help disaster recovery team members communicate with each other, with managers, and with users faster and
more effectively. Moving progress bars and green status icons can reassure you that the disaster recovery process is working as it should.

**DBA Toolkit**

The DBA Toolkit is a suite of useful tools to simplify common DBA tasks and make them faster, easier, and more accurate to perform.

For more information, see “DBA Toolkit functions and tools” on page 62.

**Automated IMS definitions**

BMC database administration products for IMS provide automated discovery of IMS components to streamline the process of defining your IMS environment in the console.

The Auto Configure wizard guides you through a series of pages that are already populated with information from your IMS control region and DBRC. You make simple selections, and the wizard defines the IMSPLEX, populates the databases, ddnames, and data sets for the IMS subsystem, adds database groups, and registers all databases in the IMSPLEX.

**Built-in IMS experience**

Advisor was developed by a team of IMS database experts at BMC and incorporate the knowledge that has been gained through years of IMS experience.

The BMC goal was to deliver a database maintenance strategy for all levels of DBAs. While automation features help experienced DBAs be more productive, less-experienced DBAs can start using the products immediately and can learn sound data management practices at the same time.

**Self-managing product components**

Advisor is essentially self-managing. Aside from implementing access controls, the only other administration tasks are related to managing the data that is stored in the repositories. The management of repository data can be automated.
You can set schedules to back up the repositories, purge repository data, and synchronize repository data with up-to-date database registry information. The size of repositories is adjusted automatically to use space effectively.

**Outstanding customer support**

BMC backs its products with outstanding customer support.

Expert help is always on call. You can speak directly with a support specialist who has years of experience with IMS database maintenance. You can e-mail Customer Support, and you can obtain technical support on the Customer Support web page or by telephone 24 hours a day, 7 days a week.
Getting started with Advisor and DBA Toolkit functions

This section introduces the major tasks for installing, implementing, and using functions and features of the BMC database administration products for IMS in the console.

Information requirements

Before you read this chapter, you should have basic knowledge of Database Advisor, Recovery Advisor, and the DBA Toolkit.

You can obtain this knowledge by reading “Getting to know Advisor and DBA Toolkit functions” on page 33.

This chapter is a guide for starting tasks; it does not provide detailed concepts, instructions, or reference material. You can follow the cross-references in this chapter to other chapters that contain detailed information about the tasks. Or you can start using the interface immediately with the aid of the online Help.

Note

Database Advisor, Recovery Advisor, and the DBA Toolkit share common architecture and components. For simplicity, the sections about installing, configuring, customizing, and managing mainframe components, the console, and IMS components use the term Advisor instead of Database Advisor, Recovery Advisor, and the DBA Toolkit when a distinction between Database Advisor, Recovery Advisor, and the DBA Toolkit is unnecessary for clarity or correctness.
Installing, configuring, and starting mainframe components

This section introduces the mainframe components and how they are installed, configured, and started. You might require this information to ensure that required elements are present and enabled for use.

Overview of mainframe components

The following figure shows the mainframe components and the relationships between them.
The following components support Advisor:

- The console is the graphical user interface (GUI). The console runs on a client workstation under the Microsoft Windows operating system and communicates with the User Interface Middleware (UIM) server through TCP/IP technology.

- The UIM server translates communication between the console and the ADV server. The UIM server runs on the z/OS™ platform.

- The ADV server manages all Advisor functions. The ADV server communicates with the UIM server, ADV repositories, IMS subsystems, and IMS database utilities. The ADV server runs on the z/OS platform.
ADV repositories store Advisor data, such as statistics about your databases, parameter values you have set, and configuration information about your environment. ADV repositories are VSAM data sets that are shared by all ADV servers in a sysplex. One ADV server is assigned to handle I/O to the repositories.

The BMC Consolidated Subsystem (BCSS) manages APF-authorized functions and performs processing for intercepted operating system requests. Advisor uses the BCSS with Cross Product Connectivity (CPC) technology. If CPC is installed with BCSS, the Advisors can share a BCSS with other BMC products. The BCSS/CPC works with the ADV server. The BCSS/CPC communicates with the BCSS/CPCs on other z/OS images through the z/OS coupling facility.

The BMC Primary subsystem (BMCP), which is not shown in the diagram, establishes supervisory services for many BMC products. The BMCP communicates in the background with the BCSS. One BMCP is installed on each z/OS image and is shared by all BMC products that are running on that image.

IMS database utilities, which are not shown in the diagram, communicate with Advisor and IMS databases. The utilities are functions that are supplied with the BMC solutions and the basic IBM reorg functions that are supplied with IMS products.

- All functions that are provided with the BMC MAXM Reorg solutions communicate statistical information to Database Advisor.
- Fast Path database utilities communicate with Database Advisor and the IMS databases. The Analyze function that is offered with the BMC Fast Path/EP Suites provides statistical information to Database Advisor.
- BMC backup and recovery utilities communicate with Recovery Advisor and the IMS databases.

Installing and configuring mainframe components

The person who installs the mainframe components uses the provided installation system to configure installation JCL.

Configured jobs accomplish the following actions:

- Create product libraries
- Install product authorization passwords that license product components for use on a CPU
- Create repositories
- Perform other installation tasks
The installer also ensures that required APF authorization is performed for product components.

When installation is complete, the installer configures the BCSS/CPC, the BMCP, the UIM server, and the ADV server, and ensures that the appropriate system access facility (SAF) security is established.

To activate Fast Path DEDB support in Database Advisor, you must install the BMC Fast Path/EP load library.

For complete information about installing and configuring mainframe components, see the Installation System documentation.

### Starting subsystems and servers

To enable use of Advisor and the console, started tasks for required subsystems and servers must be started.

The start (S) command must be issued for the BMCP, the BCSS/CPC, and the UIM server.

The ADV server is started automatically during initialization of the BCSS/CPC. For example, the following commands are issued:

```
S BMCP
S MXAP
S MXAPUIM
```

For complete information about starting and administering subsystems and servers, see the Installation System documentation.

### Starting to work with the console

The console is the interface for Advisor.

This section introduces and summarizes the tasks that you perform to install and launch the console, connect to a host system, and work with the console. “Installing and using the console” on page 95 provides complete information.
Installing the console

The console client is installed on each machine from which you want to use Advisor. You connect to the UIM server (which must be active) through a supported web browser. The UIM server displays a web page that you use to download and set up a wizard for installing the client.

The wizard downloads the client files, installs the client, and places a shortcut on the program startup menu that you can use to start the console. The wizard can also launch the console at the end of the installation process. No system restart is needed.

For details, see “Installing the console” on page 97.

Installation of updates to the console is performed in a similar way as an initial installation. When you launch the console, the UIM server determines whether console updates are available. If updates are available, the UIM server displays the web page from which you can download and run a wizard to update the client files.

Launching the console

Use the following procedure to launch the console from the program startup menu.

1. From the Start menu, choose Programs => BMC Software => Database Management Console.

2. If the host selection dialog box is displayed, select a host and click Go.

3. In the logon window, enter your user ID and password, and if necessary, your group name and account number.

   The console (Overview of the console on page 107) is displayed.

   For more information, see “Launching and exiting the console” on page 104.

Connecting to a host

When you install and launch the console, and before you can work with IMS objects, you must connect to a host system through a defined connection.

A defined connection typically is created once by an administrator and thereafter is available to all users. You can set up your client to connect to a host automatically. If no defined connections are available when you launch the console, a dialog box prompts you to create a defined connection.
For more information, see “Using the Add Host Connection command” on page 111.

Using the console

The console follows typical Windows conventions.

Figure 10: Console

The Navigation window contains the object tree, which is a hierarchical list of the objects that are identified to Advisor in the console. The tree is similar to a directory of files in a folder. To interact with an object on the tree, right-click the object and choose an option from the pop-up menu.

The results of your selection are displayed in the work area to the right of the Navigation window. You can navigate among multiple open windows in the work area by clicking the buttons along the bottom edge of the work area.

Messages about the console session are displayed in the Messages area below the Navigation window.
For more information about navigating and working with the console, see “Overview of the console” on page 107.

**Identifying IMS components**

For Advisor to work with IMS components on a host system, your IMS environment must be identified. Advisor works with IMS databases collectively in database groups.

This section summarizes how IMS environments and database groups are defined. “Identifying and using IMS components” on page 115 provides complete information.

**Defining your IMS environment**

Your IMS environment must be defined to Advisor so that it can recognize the IMS components in your environment, collect data about them, analyze them, and display them in the console.

When the IMS environment has been defined on a host system, all users can view the defined IMS component objects on the object tree.

Defining the IMS environment on a host system typically is performed one time for the environment through the Auto Configure wizard. The wizard reduces the time and effort that are required for this task by automatically discovering components in the environment. Components can be added and changed manually as necessary.

To define your IMS environment, perform one of the following steps in the Navigation window:

- To start the Auto Configure wizard, right-click the **IMS Databases** object for the host connection and choose **Configuration => Auto Configure**.

- To add an IMS component, right-click the object to which the new component will be a subordinate object and choose **Configuration => Add objectType**. For example, if you are adding a database group to an IMSPLEX, right-click the IMSPLEX and choose **Configuration => Add Database Group**.

For more information, see “Defining your IMS database administration environment” on page 115.
Defining database groups

Advisor works with groups of related databases to simplify database maintenance tasks and to make your actions efficient and effective. The database group can be based on criteria such as the DBRC change accumulation group name, the DBRC database data set group name, and the DBD name.

Database groups can be created during the Auto Configure wizard process. For more information, see “Starting the Auto Configure wizard” on page 120.

Database groups can also be created manually.

To start creating database groups manually, right-click the IMSplex that you want to contain the database group and choose Configuration => Add Database Group.

For more information, see “Adding groups manually from RECONs” on page 127.

A single database can belong to multiple database groups. However, only one group can own the database. If parameter changes are made to a group, database ownership prevents those changes from being inherited by databases that are owned by a different group. To customize a parameter for a database that belongs to more than one database group, you must select the database object that is displayed under the owning database group on the tree.

To start assigning or transferring database ownership, right-click the database group that contains the databases for which you want to manage ownership and choose Configuration => Manage Database Ownership.

For more information, see “Managing database ownership” on page 133.

Searching for databases

You can use the database search feature to locate a database or several related databases across an IMSplex.

The search results are displayed in the Navigation window as a temporary list of databases that meet the search criteria (DBD name and partition or area name).

To start searching for databases, right-click an IMSplex and choose Search Databases.

For more information, see “Searching for databases” on page 136.
Customizing parameters and schedules

You can control Advisor processes by customizing parameter values and process schedules.

This section introduces parameters and schedules. “Customizing parameters and schedules” on page 225 provides complete information.

Configuring JCL parameters

Advisor requires that you provide values for the job control parameters that are used to generate JCL.

These parameters have no defaults because they typically are specific to your organization or site.

You can use the Configure JCL Parameters wizard to provide these values. To start, right-click the IMS Databases object or an IMSPLEX object in the Navigation window and choose Configuration => Configure JCL Parameters.

For more information, see “Configuring JCL parameters” on page 225.

Editing parameters

Advisor is distributed with default values (at the host-system level) for most parameter values. These default values are inherited by all subordinate objects unless you customize the values for a subordinate object.

To start editing parameters for an object, right-click the object in the Navigation window and choose Configuration => Maintain Parameters.

The Maintain Parameters window is displayed. The left side of the Maintain Parameters window contains a list of parameter categories, such as Thresholds, Goals, Analysis, and Job Control. When you select a parameter category, the parameters in that category are displayed on the right side of the window.

For more information, see “Editing parameters” on page 229.
Editing schedules

You use schedules to control the execution of functions that automate collection processing and repository management. You can control the frequency of function execution by editing the corresponding schedule.

To start editing the schedule for all functions other than the Repository Backup function, right-click an IMSPLEX object in the Navigation window and choose Configuration => Schedules. To start editing the schedule for the Repository Backup function, right-click the host-system level in the Navigation window and choose Configuration => Schedule Repository Backups.

For more information, see “Editing schedules” on page 285.

Handling exceptions

An exception represents an existing or potential problem that Database Advisor or Recovery Advisor has identified for an object (which is a database, change accumulation, log, or RECON).

This section introduces the process of handling detected exceptions. “Handling exceptions” on page 295 provides complete information.

Working with the Exception List

If Advisor detects an exception for an object, the object is listed on the Exception List.
The Exception List displays all exceptions that were found for a host system, an IMSPLEX, or a database group.

Figure 11: Exception List window

To display the Exception List, right-click an object and choose Exception List.

The Summary table at the top of the Exception List contains a row for each object that has one or more exceptions. An icon in the **Highest Status** column indicates the status of the most severe exception that has been reported for the object:

- The Warning (yellow) icon indicates an exception that might not need to be corrected immediately or that an exception is predicted but has not yet occurred.
- The Critical (red) icon indicates that a threshold has been met or exceeded and requires corrective action.
- The Dead (black) icon indicates that the database has reached an operating system limit and requires immediate action.

For more information, see “Working with exceptions and the Exception List” on page 303.
Working with exceptions

To list the individual exceptions that have been detected for an object, select the object row in the Summary table.

Each exception is listed for the selected object in a row of the Detail table at the bottom of the Exception List. The specific exception type is identified in the Description column.

For more information, see “Exception types” on page 296.

Working with solutions

A solution is a set of job steps that execute database utilities and other processes. You can generate a solution to correct all or selected exceptions for an object.

To display a list of the solutions that you can use to correct all (or selected) exceptions, right-click a row (object) in the Summary table of the Exception List and choose Solution or select a row and click Solutions.

To generate JCL immediately for the recommended solution, right-click a row in the Summary table of the Exception List and choose Recommended JCL.

Generated JCL is displayed on the JCL tab in the Solutions window. You can edit, submit, save, or clear the displayed JCL.

For more information, see “Using solutions from the Exception List” on page 333.

Analyzing a database exception

If you have questions about the exceptions that are reported for a database, you can select the exception and view information about the database, such as base values and historical information.

You can also view and edit parameters, such as site values, thresholds, and goals.

To access menu commands that help you analyze an exception, right-click a row in the Detail table of the Exception List and choose the command.

For more information, see “Analyzing Database Advisor exceptions” on page 162.
Using the Solution batch utility

You can use the Solution batch utility (program BRIUBSOL) to automate the process of generating JCL for exceptions that are reported on the Exception List.

For example, if you schedule the solution batch utility to run every night, optimized JCL is generated and is ready to submit the next morning.

For more information, see “Handling exceptions with the Solution batch utility” on page 341.

Controlling conditional processes

The Conditional Image Copy feature of Backup and Recovery Solution for IMS saves time and resources by bypassing a scheduled batch image copy of a database, area, or partition.

The copy is bypassed unless updates have occurred for the object or an image copy is required. The Conditional Reorg feature of Database Advisor saves time and resources by bypassing a scheduled reorganization of a full-function database or HALDB unless Database Advisor has flagged the database as requiring reorganization.

This section introduces the tasks for implementing and using conditional processes. “Controlling conditional processes” on page 385 provides complete information.

Controlling the IC Triggering by CA feature

You can use the console to activate the IC Triggering by CA feature and specify options for running the feature automatically.

1 In the Navigation window, right-click the object (host, IMSplex, or CA group) for which you want to activate the IC Triggering by CA feature, and choose Configuration => Maintain Parameters.

2 From the tree in the left pane of the Maintain Parameters window, select the IC Triggering by CA branch.

3 Edit and save the parameters.

4 Perform one of the following actions for the BMC Change Accumulation utility:
In a global options module that applies to the Change Accumulation utility job step, enter \textbf{Y} in the \textbf{IC Triggering by CA} field.

In the CAPSYSIN control statement data set, specify the \texttt{SMARTCA(Y)} keyword.

5 Perform one of the following actions for the BMC Image Copy utility:

- In a global options module that applies to the Image Copy utility job step, enter \textbf{Y} in the \textbf{IC Triggering by CA} field.

- In the ICPSYSIN control statement data set, specify the \texttt{SMARTCA(Y)} keyword.

For more information, see “Using the IC Triggering by CA feature” on page 385.

\section*{Controlling the Conditional Image Copy feature}

You can use the console to activate the Conditional Image Copy feature and specify options for running the feature automatically.

1 In the Navigation window, right-click the object for which you want to activate the Conditional Image Copy feature and choose \textbf{Configuration => Maintain Parameters}.

2 In the Maintain Parameters window, select Conditional IC in the parameter list.

3 Edit and save the parameters.

4 Perform one of the following actions for the Image Copy utility:

- In a global options module that applies to the Image Copy utility job step, enter \textbf{Y} in the \textbf{Conditional Image Copy} field.

- In the ICPSYSIN control statement data set, specify the \texttt{SMARTIC(Y)} keyword.

For more information, see “Using the Conditional Image Copy feature” on page 390.

\section*{Controlling the Conditional Reorg feature from the console}

You can use the console to activate the Conditional Reorg feature and specify options for running the feature automatically.
1 In the Navigation window, right-click the object for which you want to activate the Conditional Reorg feature and choose Configuration => Maintain Parameters.

2 In the Maintain Parameters window, select Conditional Reorg in the parameter list.

3 Edit and save the parameters.

For more information, see “Using the Conditional Reorg feature” on page 397.

Using the stand-alone Conditional Reorg utility

You can implement the Conditional Reorg feature by modifying a reorganization job to add a job step that executes the stand-alone Conditional Reorg utility before the first reorganization job step.

**Figure 12: Stand-alone conditional reorganization JCL**

```jcl
/*
//* CONDITIONAL REORG STEP
//*
//BMCCOND EXEC PGM=BMCCOND
//STEPLIB DD   DISP=SHR,DSN=YOUR.ADVISOR.LOADLIB
//PLUSIN DD    *
ADVISE DBD(YOURDBD) RECON1(YOUR.RECON1.DSNAME)
//*
//* BYPASS THE REORG STEP IF NO REORG IS NEEDED FOR YOURDBD
//*
//REORG EXEC BMCREORG,COND=(0,NE,BMCCOND)
```

For more information, see “Using the stand-alone Conditional Reorg utility” on page 401.

Monitoring recovery job steps

If you have a license for the Backup and Recovery Solution for IMS product, you can use the Monitor Recovery function to watch the progress of recovery job steps, determine whether a problem has occurred, and check on recovery of specific objects.

The Monitor Recovery function receives information about recovery processing from the BMC Recovery utility and displays this information in the Monitor Recovery window (Figure 13 on page 87) and in dependent windows in the console. For more information, see “Monitoring recovery job steps” on page 409.
To access the Monitor Recovery window, right-click an IMS RECON object in the Navigation window, and choose Monitor Recovery from the pop-up menu.

**Figure 13: Monitor Recovery window**

---

**Performing common database administration tasks**

The DBA Toolkit is a collection of functions and features that reduce the time and effort for performance of common database administration tasks.

**Generating JCL**

The following table lists and describes the options you can use to generate JCL.
### Table 2: JCL generation options

<table>
<thead>
<tr>
<th>To do this:</th>
<th>You can use this:</th>
<th>For more information, see:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate optimized JCL that executes database administration utilities and processes for database, partition, area, change accumulation, and log objects</td>
<td>Console&lt;br&gt;In the console, right-click the RECON, database, or other object for which you want to generate JCL, and choose Generate JCL.&lt;br&gt;The Generate JCL window displays JCL scripts that are available and appropriate for the object. BMC provides predefined JCL scripts to execute many utilities and processes.</td>
<td>“Generating JCL from the Generate JCL window” on page 338</td>
</tr>
<tr>
<td>Generate JCL in batch</td>
<td>JCL Generation utility (program CPCBATCH with the GENJCL command)&lt;br&gt;The JCL Generation utility creates JCL for an object and for a combination of solutions, classes, and features.</td>
<td>“Generating JCL with the JCL Generation utility” on page 350</td>
</tr>
<tr>
<td>Write custom JCL scripts for your environment</td>
<td>The JCL scripting language BMC provides&lt;br&gt;“Using the JCL script language” on page 499</td>
<td></td>
</tr>
<tr>
<td>Convert JCL written for non-BMC products to JCL compatible with BMC database products for IMS</td>
<td>Utility Migration JCL Generator&lt;br&gt;Generating JCL with the Utility Migration JCL Generator on page 359</td>
<td></td>
</tr>
</tbody>
</table>

### Generating recovery JCL

You can use the console to generate recovery JCL for all objects in a database group. The Recovery Wizard prompts you for the required information and specifications, generates the JCL in the background, and notifies you when the JCL is created. You can access and work with the JCL in the Workflow Status window.

To generate recovery JCL, right-click the database group to be recovered, and choose **Recovery Wizard**. To open the JCL Generation Status window, click the link in the notification popup dialog, or right-click the IMS Databases object and select **Workflow Status**.

For more information, see “Generating and working with recovery JCL” on page 370.
Viewing job output

The Jobs perspective provides the convenience of viewing the JES log through the console. Jobs in the Jobs perspective are not limited to IMS jobs or jobs that are submitted through the Generate JCL function.

To open the Jobs perspective, click the Jobs tab.

For more information, see “Viewing jobs in the console” on page 372.

Using historical reports

The DBA Toolkit generates reports that help you stay informed about your databases. You can use these reports to evaluate the current state of your databases, to research exceptions, and to track trends.

State reports offer a detailed, comprehensive view of the current state of your IMS data. You can view state reports for database, partition, area, and data set objects. If you have a license for Database Advisor, current state reports can also be used to chart a data forecast that graphically represents the comparison of the current state of a key element to the threshold value that is defined for the key element.

History reports display data in graphs and tables that you can customize to control the type of data in the report. You can view historical reports for database, partition, area, data set, and segment objects. You can also export tabular data for use in third-party spreadsheets.

Compare History reports display historic information about multiple databases and partitions so that you can visually compare the information in a graph or table format.

Job History reports provide historic information about utility jobs that have executed against IMS databases in the selected system. You can access job history reports at the host-connection level and the IMSPLEX level.

To view reports, in the Navigation window right-click the object for which you want to view a report and choose Reports.

For more information, see “Viewing state reports for database-type objects” on page 431 and “Viewing history reports” on page 438.
Disassembling IMS control blocks, and mapping DBDs

The DBA Toolkit disassembly tool provides a quick and convenient way to produce source code for the following types of IMS control blocks:

- Database definition (DBD)
- Program specification block (PSB)
- Application control block (ACB)
- Dynamic allocation (MDA)
- Message format service (MFS); a MAXM Database Advisor license is required for this type

To disassemble IMS control blocks, perform one of the following steps:

- To disassemble a DBD, PSB, ACB, or MDA control block at the IMSPLEX level, in the Navigation window right-click the IMSPLEX object and choose **Disassemble => DBD/PSB/ACB/MDA**.

- To disassemble a DBD at the database level, right-click the database object and choose **Disassemble**.

- To disassemble an MFS control block, in the Navigation window right-click the IMSPLEX object and choose **Disassemble => MFS**.

After disassembly, you can save the source code (except for MFS source code) to your personal computer or to a partitioned data set on the mainframe.

If you disassembled a DBD, you can also view a DBD map by clicking the DBD Map tab. A DBD map is a graphical representation of the segment hierarchy, internal logical relationships, and secondary properties of the DBD.

For more information, see “Disassembling DBD, PSB, ACB, and MDA control blocks” on page 464 and “Creating and viewing DBD maps” on page 472.

Migrating to HALDBs and managing HALDB partitions

MAXM Database Advisor offers interactive HALDB modeling and migration tools to help you migrate traditional full-function databases to high availability large databases (HALDBs) and manage HALDB partitions.

These tools help you select hierarchical indexed direct access method (HIDAM) key ranges or hierarchical direct access method (HDAM) parameters for a migration. You can also use the tools to model the effects of a partition selection exit (PSE) for setting up a new HALDB and for splitting or rebalancing existing HALDB partitions.
Using your proposed specifications and information extracted from an existing database, the tools generate sample JCL, control statements, and definitions for the selected task.

To get started with the HALDB modeling and migration tools, right-click an object (an IMS Plex, a full-function database, HALDB, or HALDB partition) and select the tool you want to use.

For more information, see “Using HALDB modeling and migration tools” on page 472.

**Tuning partitions and randomizers**

MAXM Database Advisor offers the batch partition and randomizer tuning tool to help you optimize data storage in IMS full-function databases. You can also use this tool in a similar way as the HALDB modeling and migration tools to migrate traditional full-function databases to HALDBs.

For more information about the batch partition and randomizer tuning tool, see “Using the batch partition and randomizer tuning tool” on page 476.

**Calculating space for a DEDB area**

The DBA Toolkit DEDB space calculation tool offers a quick and convenient way to calculate disk space requirements for a Fast Path data entry database (DEDB) area before allocating space.

The DEDB space calculation tool provides the following information:

- Percentage breakdown of total space used
- Graphic depiction of area space distribution
- Breakdown of section sizes by bytes, CIs, and percentage of area
- Detailed information about area initialization

To calculate space for a DEDB area, in the Navigation window right-click the RECON object for which you want to calculate DEDB area space and choose **DEDB Space Calc**.

For more information, see “Calculating DEDB space” on page 488.
Annotating objects

Use the Notes window to enter free-form information about the selected object (group, database, partition, or DEDB area).

In the Navigation window, right-click the object for which you want to enter notes, and choose Notes.

For more information, see “Annotating objects” on page 497.

Managing product components

This section briefly describes when, why, and how the product administrator manages Advisor components.

For more information about these topics, see the Database Products for IMS Customization Guide.

Controlling user access

With BMC database administration products for IMS, you can monitor test and production databases and take actions on databases when exceptions occur.

For various purposes (such as auditing and training), you might want to limit users’ ability to access and use the products. You can limit the IMS systems in which users can use the products, and you can also limit the product functions that they can perform in an individual IMS system.

To limit access, you use the system authorization facility (SAF). SAF is part of z/OS and provides the interface to your security product such as RACF, ACF2, or Top Secret. SAF tracks z/OS resources and uses security rules to determine who can access these resources and the type of access that they have (such as read or update access).

Your security administrator identifies the resources to be managed within the FACILITY class and identifies the users and their permitted type of access to each resource. Based on a user’s permission to access those SAF resources, the user interface enables or disables the corresponding functions.

You might want to define access controls to implement and enforce your site’s security policy. Access controls are typically established immediately after product installation but can be implemented later. Implementation will require cooperation from your security administrator and your senior DBA.
By default, product functions and features are unrestricted. If you want to limit a user’s ability to use these functions and features, you must implement access controls.

Managing repository data

Advisor repositories are essentially self-managing. You can automate repository backups, data purges, and database registration synchronization by using scheduled functions. By default, all functions are set to execute daily.

To edit the schedule for backing up repositories, right-click a host object in the Navigation window and choose Configuration => Schedule Repository Backups.

Cloning repository data

You can use the Advisor repository cloning process to duplicate parameters (threshold values, goals and site values) for multiple IMSPLEXes which have similar database or partition structures.

The cloning process offers the following methods for using established parameters:

- You can clone data within the same Advisor system to a new IMSPLEX.
- You can export data to a sequential data set and then import the sequential data to a different Advisor system and IMSPLEX.
Installing and using the console

This section contains detailed information about installing and using the console, which is the graphical user interface that is used with Database Advisor, Recovery Advisor, and the DBA Toolkit. When the UIM server is installed and running, you can install, update, and launch the console.

Information requirements

Before you read this section, you should have a high-level knowledge of the architecture of the functions and features of the BMC database administration products for IMS in the console.

This architecture is explained in “Installing, configuring, and starting mainframe components” on page 72. You should also have a basic understanding of how to use applications that work under the Microsoft Windows operating system. You can obtain this knowledge by reading relevant documents in the Windows information set from Microsoft.

If you have used the console with another BMC product, you can skip most of the information in this chapter. The basic features of the console work in the same way, regardless of which product you are using. However, some advanced features are not available for all BMC products.

Note

If you want to use BMC products from different product families in the same console, contact BMC for more information.

After you have installed the console and know how to use it, you can go to “Identifying and using IMS components” on page 115 to learn how to define your IMS components to Advisor. Or you can start exploring the console with the aid of the online Help.
Verifying personal computer requirements

The following table lists the minimum requirements for installing the console on a personal computer.

Table 3: Personal computer requirements

<table>
<thead>
<tr>
<th>Resource</th>
<th>Minimum requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>You will need one of the following operating systems:</td>
</tr>
<tr>
<td></td>
<td>■ Windows 7 or later</td>
</tr>
<tr>
<td></td>
<td>■ Windows Server 2008 or later</td>
</tr>
<tr>
<td>TCP/IP port 3683</td>
<td>Ensure that TCP/IP port 3683 is available for console and UIM communications.</td>
</tr>
<tr>
<td>Memory</td>
<td>256 MB RAM (512 MB recommended)</td>
</tr>
<tr>
<td>Video resolution</td>
<td>XGA (1024 x 768) or higher resolution</td>
</tr>
<tr>
<td>Network</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>Web browser</td>
<td>Most standard Web browsers</td>
</tr>
</tbody>
</table>

Configuring the console

After the UIM server is installed and running, you can install, configure, and update the console. The UIM server downloads the required code to the console computer.

The default installation location is determined by the computer's operating system. For example, the default installation location could be C:\Users\user\Program Files\BMC Software\Database Management Console.

User requirements

The console is installed once on each machine.

If the console is installed on a computer on which you have administrator privileges, the console can be shared by all users.

If the console is installed on a machine on which you do not have administrator privileges, you can install a single instance of the console for your use.
Privilege type | Type of installation
--- | ---
Administrator | Install one console  
The default location is C:\Program Files\BMC Software\Database Management Console.

Non-administrator | Every user can install one console  
The default location is C:\Documents and Settings\user\My Documents\BMC Software\Database Management Console.

### Installing the console

Use the following procedure to install the console on a personal computer.

**Before you begin**

You will need:

- The name of the host computer on which the UIM server is running
- The UIM server’s port number
- Access to a supported web browser
  
  If you do not know the host name and port number, consult your system administrator.

- If you are implementing Transport Layer Security (TLS) before installing the console, you need to have completed the implementation procedure described in “Implementing TLS” on page 99.

**To install the console**

1. From your web browser, enter the URL for the UIM server on the mainframe.

   For example, enter [http://uimServerHostName:uimPortNumber/dna/index.html](http://uimServerHostName:uimPortNumber/dna/index.html), replacing the variables with your information.

2. On the resulting web page, click **Install Local Client**.

3. Based on the browser that you are using, either open the file from its current location, or save the file to your hard drive and then run the program:

   - If you are using Internet Explorer, perform the following steps:
1 In the File Download dialog box, select **Run this program from its current location**, and click **OK**.

2 If the Security Warning dialog box is displayed, click **Yes**.

- If you are using Mozilla Firefox or a similar browser, clicking **Install Local Client** downloads the .exe file to the browser's default downloads folder. Browse to the saved file, and double-click it.

4 Enter the requested information in the BMC Database Management Wizard.

5 On the last page of the wizard, select **Launch the Database Management Console**.

6 Click **Finish**.

   The BMC Database Management Console starts.

7 On the last page of the wizard, select **Launch the Database Management Console**.

8 Click **Finish**.

   The BMC Database Management Console starts.

If you implemented TLS prior to installing the console, a pop-up might be displayed indicating that the certificate is not trusted. This can occur if the certificate is self-signed, the intermediate certificate authority certificates are not in the RACF key ring, or the root certificate authority certificate is not in the trust store.

a  *(optional)* Accept the certificate.

b  Cut and paste the provided Uniform Resource Locator (URL) into your browser's address bar and press **Enter**.

   A pop-up is displayed that provides the URL.

   The BMC Database Management Console starts.

**Where to go from here**

---

**Tip**

You can also implement TLS after you have installed the console. For more information, see “Implementing TLS” on page 99.

---

98  *Database Products for IMS Advisors and Toolkit User Guide*
Implementing TLS

To implement TLS, you need to indicate the level of encryption you want and specify the location of the server certificate and ICSF label of the private key that matches the certificate’s public key.

Before you begin

Have the following items in place:

- In IBM Resource Access Control Facility (RACF):
  - A TLS certificate
  - A certificate authorization chain
  - A key ring with the TLS certificate connected first followed by any intermediate certificates used to sign the key

- Have Integrated Cryptographic Service Facility (ICSF)
  The ICSF Private Key Data Set (PKDS) should contain the private key that matches the TLS certificate.

To implement TLS

1 Open for edit your User Interface Middleware (UIM) startup member.

The UIM startup member is an XML document that defines global operation parameters for the UIM Server.

The UIM startup member is specified as the -M member name on the EXEC statement in the UIM Server PROC. This member is in the HTPPARAM DD concatenation, usually found in the UBMCCNFG data set.

2 In the startup member, add or edit the following parameter to define the encryption level required:

   `<BMC_PARM ID="ENCRYPTION_LEVEL" VALUE="encryptValue" />`

An example of the UIM startup member is provided in “Example UIM startup member” on page 101. Use one of the following values for `encryptValue`:

- **SSL-IF**
  Uses SSL/TLS if the client supports it. This is the recommended value for an initial implementation to allow existing clients to connect.

- **SSL-REQUIRED**
  Does not accept a connection unless it is a TLS handshake. This option should be used with caution because any existing client connections will not work.
until the client code level has been reinstalled to the updated client with the TLS support. This will require a reinstall of the client and not an update.

**Note**

SSL-REQUIRED causes any legacy client and Database Performance for DB2 connections to not work. It is strongly recommended that SSL-IF be implemented and once all clients are updated, then switch to SSL-REQUIRED. If Database Performance for DB2 is being used, SSL-REQUIRED should not be implemented until support for SSL is available in Database Performance for DB2.

3. In the startup member, add the following section to define the location for the UIM Server certificate and associated private key:

```xml
<BMC_PARM ID="SSL_STORE" VALUE="certStorage">
   <BMC_PARM ID="SSL_CERTIFICATE" VALUE="owner/KeyRingName" />
</BMC_PARM>
<BMC_PARM ID="SSL_STORE" VALUE="ICSF">
   <BMC_PARM ID="SSL_PRIVATE_KEY" VALUE="icsf.private.key.label" />
</BMC_PARM>
```

An example of the UIM startup member is provided in “Example UIM startup member” on page 101. The keywords and valid values are:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Valid entries for associated VALUE keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL_STORE (first occurrence)</td>
<td>Names the storage medium for the certificate.</td>
<td>Use one of the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Storage is a sequential data set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ R_DATALIB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Storage is a RACF key ring and optionally the key ring owner</td>
</tr>
<tr>
<td>SSL_CERTIFICATE</td>
<td>Name of the certificate to be used by UIM.</td>
<td>Specify one of the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ A data set name if DSN was indicated for the associated SSL_STORE keyword</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ A RACF key ring name if R_DATALIB was indicated for the associated SSL_STORE keyword</td>
</tr>
<tr>
<td>SSL_STORE (second occurrence)</td>
<td>Names the storage medium for the private key.</td>
<td>ICSF indicates that the storage is an ICSF label.</td>
</tr>
</tbody>
</table>
4 Save and close your User Interface Middleware (UIM) startup member.

5 Repeat steps Step 1 on page 99 through Step 4 on page 101 for every instance of the UIM Server that you have.

6 Continue with one of the following procedures:

   ■ If you are implementing TLS before installing the console, install the console (see “Installing the console” on page 97.

   ■ If you are implementing TLS for a previously installed console, reinstall the console (see “Reinstalling the console” on page 102.

Example UIM startup member

The following example of the UIM startup member indicates where to change or insert the required statements:
Reinstalling the console

After implementing TLS, you will have to reinstall the console.
To reinstall the console

1 Open the console as you normally would.

A pop-up might be displayed indicating that the certificate is not trusted. This can occur if the certificate is self-signed, the intermediate certificate authority certificates are not in the RACF key ring, or the root certificate authority certificate is not in the trust store.

2 (optional) Accept the certificate.

3 Cut and paste the provided Uniform Resource Locator (URL) into your browser's address bar and press Enter.

A pop-up is displayed that provides the URL.

The console is reinstalled and opened.

After implementing and reinstalling the console, the connection icon includes an SSL indicator as shown in the following figure:

![Navigation](image)

Updating the console

When you launch the console, the files on your personal computer (local client) are compared to the file residing on the UIM server. If a change in the installed products has occurred, the console is updated automatically.

**Note**

If the program that installs the console on the UIM server is updated, you must enter the URL for the UIM server on the mainframe to update the console. For more information, see “Installing the console” on page 97.
Uninstalling the console

Use the following procedure if you need to remove the console from your personal computer.

To uninstall the console

1. In Microsoft Windows, open the Control Panel.
   For more information, see the documentation for your version of Windows.

2. In the Control Panel window, click Programs and Features.

3. Select and uninstall BMC Database Management Console (user).
   The value of user is Admin if the console was installed by an administrator; otherwise, the value is the relevant user name.

Launching and exiting the console

The console uses z/OS authentication. When you launch the console, you must provide a valid SAF user ID and password. The security administrator for your site manages the SAF account information for users.

To launch the console

1. From the Start menu, select All Programs => BMC Software => Database Management Console.
   The host selection dialog box (Figure 14 on page 104) or the logon dialog box (Figure 15 on page 105) is displayed.

Figure 14: BMC Database Management Console host selection dialog box
2 If the host selection dialog box (Figure 14 on page 104) is displayed, in the Host list, select your UIM host name.

**Tip**

You can manage the host list by clicking Manage Client Installs. For more information, see “Managing BMC Database Management clients” on page 105.

The logon dialog box (Figure 15 on page 105) is displayed.

3 In the logon dialog box, enter the user ID and password that is valid on the selected host (z/OS) system.

4 *(optional)* Enter your group and account information.

5 Click **OK**.

**To exit the console**

1 Click **userID => Logout**.

2 On the Quit? dialog box, click **Yes**.

---

**Managing BMC Database Management clients**

When you launch the console, you can manage your BMC Database Management clients from the host selection dialog box.
To manage the BMC Database Management clients

1. From the Start menu, select All Programs => BMC Software => Database Management Console.

   The BMC Database Management Console host selection dialog box is displayed.

2. Click Manage Client Installs.

   The Manage UIM Server Clients dialog box is displayed (Figure 16 on page 106).

   **Figure 16: Manage UIM Server Clients dialog box**

3. Take the appropriate actions to manage your host list:

   - To add a host, click **Add** and complete the fields in the Details area.
   - To edit a host, select it in the host list, click **Edit** and change the fields in the Details area as needed.
   - To delete a host, select it in the host list, and click **Delete**.
   - To define a host to display as the default host when logging into the console, click the check box next to the host name.
Overview of the console

The BMC Database Management Console is a Windows application. You can use it to access console-enabled BMC products through a single interface.

Figure 17: BMC Database Management Console

The console interface features the following windows:

- **Navigation window**
- **Messages window**
- **Work area**

**Navigation window**

The Navigation window displays the data sources to which you have defined connections and other related objects.

To access commands from a pop-up menu, right-click an object in the tree.
Message windows

You can view messages in the Display Messages window, and in the Messages window at the bottom of the console. When you press **CTRL-M**, the Display Messages window appears as a separate window. These message windows list messages that the product generates during the current session.

The most recent message is displayed at the top of either window. To view detailed Help, click any message number.

Table 4 on page 108 describes the information that is displayed in the Display Messages and Messages windows.

### Table 4: Message windows

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>Type of message: informational, warning, or error</td>
</tr>
<tr>
<td>Source</td>
<td>Where the message originated</td>
</tr>
<tr>
<td>When</td>
<td>Time that the message was generated</td>
</tr>
<tr>
<td>Message</td>
<td>Numerical identifier for the message and the message text</td>
</tr>
</tbody>
</table>

Message icons (Table 5 on page 108) indicate the severity of a message.

### Table 5: Message icons

<table>
<thead>
<tr>
<th>Display Messages window icon</th>
<th>Messages window icon</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td>Informational</td>
<td>Status of your system or product</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td>Warning</td>
<td>Noncritical problem that might interfere with system or product processes</td>
</tr>
<tr>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td>Error</td>
<td>Critical problem that might interfere with system or product processes</td>
</tr>
</tbody>
</table>

From the Display Messages window, you can:

<table>
<thead>
<tr>
<th>Task</th>
<th>Click</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save the messages as an XML file</td>
<td><img src="image7.png" alt="Image" /></td>
</tr>
<tr>
<td>Clear all messages</td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
<tr>
<td>Filter messages according to severity and source</td>
<td><img src="image9.png" alt="Image" /></td>
</tr>
</tbody>
</table>
To access Help or clear all messages from the Messages window, right-click a message.

### Work area

Use the work area to work with the data sources that you select from the Navigation window, or with console tools.

The work area contains a window for each item that you select, as indicated by the windows tabs at the bottom of the work area (Figure 17 on page 107). Clicking a tab lets you see the information on that tab.

**Tip**

When you finish using a window, close it. Open windows consume computer memory.

### To sort columns in the work area

1. To sort a column, click the column heading.

   An up or down arrow in the column heading shows whether the sort is ascending (up arrow) or descending (down arrow).

### To select rows in the work area

1. To select rows, perform one of the following actions:

   - To select contiguous rows, click anywhere within the row and drag the mouse over the rows.

   - To select noncontiguous rows, press **CTRL** while selecting each row.

### Selecting user options

The Set Options dialog box, which you open by choosing **userID => Tools => Options**, provides the ability to show the taskbar buttons in the work area.

You can display taskbar buttons for windows that are open in the work area. Taskbar buttons are displayed along the bottom edge of the work area. The default is to show the buttons.
Setting up connections

When you launch the console, you must define at least one host connection. After defining a host connection, you can work with resources on that host. The connection definition remains available whenever you start the console and log on.

In the IMS and DB2 perspectives of the console, the names of all defined connections are listed in the Connections folder in the Navigation window.

In the Jobs and Data Sets perspectives, you can define filters that determine which jobs or data sets are displayed. When you define a filter, you specify the connection on which to perform the filter. You can define more than one filter, and specify the same connection for all of them. For more information about adding filters to the Jobs or Data Sets perspectives, see the console Help.

In the UIMCommands perspective, you can specify which UIM servers appear as filters in the perspective by specifying the connection. The connection you specified when you logged on to the console appears by default as a filter (and not as a host connection) in the UIMCommands perspective.

Overview of the Enterprise List and personal connections

Host connections for individual users are managed separately from host connections for the entire enterprise. This separation makes it easier to isolate activities in different environments (such as testing systems versus production systems or application systems versus other application systems).

The console uses a shared Enterprise List of connections. This list resides on the UIM server. Users who have the appropriate security authority can add, delete, and edit connection information in the Enterprise List. When any user launches the console, the Enterprise List identifies the host connections that are defined.

Each user has a personal list of connections. You can define a connection in your personal list manually by entering connection information (such as the host name and port number). You can also add connections to your personal list by selecting predefined connections from the Enterprise List. After you define a host connection in your personal list, that connection definition remains available whenever you start the console and log on.

Managing the Enterprise List

If you have the appropriate authority, use the following procedure to add, edit, or delete connections in the shared Enterprise List on the UIM server.
To manage the Enterprise List

1 In the Navigation window, right-click the Connections folder and select Manage Host Connections in the Enterprise List.

The Manage Enterprise Connections dialog box is displayed.

2 Add, edit, or delete a connection as follows:

<table>
<thead>
<tr>
<th>To</th>
<th>Perform these steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a connection</td>
<td>1 Click <strong>Add</strong>.</td>
</tr>
<tr>
<td></td>
<td>2 Enter the connection information in the Details area.</td>
</tr>
<tr>
<td></td>
<td>3 Click <strong>Done</strong>.</td>
</tr>
<tr>
<td>Edit a connection</td>
<td>1 Select the connection, and click <strong>Edit</strong>.</td>
</tr>
<tr>
<td></td>
<td>2 Change the connection information in the Details area.</td>
</tr>
<tr>
<td></td>
<td>3 Click <strong>Done</strong>.</td>
</tr>
<tr>
<td>Delete a connection</td>
<td>1 Select the connection.</td>
</tr>
<tr>
<td></td>
<td>2 Click <strong>Delete</strong>.</td>
</tr>
</tbody>
</table>

3 Click **OK** to apply your changes to the shared Enterprise List and close the dialog box.

Using the Add Host Connection command

To define a single connection quickly, use the **Add Host Connection** command. Use this method to create a connection definition when you know the required connection information, such as host and port.

To add a connection

1 In the Navigation window, right-click the Connections folder and click **Add Host Connection**.

The Add Connection dialog box is displayed.
2 In the **Host** box, enter the name of the host.

3 In the **Port** box, enter the UIM port number.

The value in the **Display Name** field is generated automatically from the host name and the port number.

4 *(optional)* In the Description box, enter a descriptive name for the connection.

5 In the Login Information area, select whether to connect by using console credentials or with credentials that you specify.

---

**Note**

Console credentials are the user name and password that you used to log on to the console. If you select **Connect using these credentials**, you must enter a user name and password. If necessary, you can also enter the account and group to which the user name belongs.

---

6 *(optional)* To connect automatically whenever the console starts, select **Autoconnect at console start**.

7 Click **OK**.

The new host connection is displayed in the Connections folder in the Navigation window.

**To connect to a host**

1 In the Navigation window, right-click the host connection and click **Connect**.

The Connect dialog box displays the connection information for the selected host.

2 In the Login Information area, select whether to connect by using console credentials or with credentials that you specify.

---

**Note**

Console credentials are the user name and password that you used to log on to the console. If you select **Connect using these credentials**, you must enter a user name and password. If necessary, you can also enter the account and group to which the user name belongs.

---

3 *(optional)* To connect automatically whenever the console starts, select **Autoconnect at console start**.

4 Click **OK**.
Adding IMS objects

When you have connected to a host, you can add IMS objects to the IMS Databases and IMS Subsystems nodes for that host.

For more information, see “Defining your IMS database administration environment” on page 115.
This section contains detailed information about defining your IMS database administration environment and working with IMS components in the console.

Defining your IMS database administration environment

You must define your IMS database administration environment in the console so that BMC database administration products for IMS can recognize the IMS components in your environment, collect data about them, analyze them, and display them in the console.

You can define your IMS database administration environment by using automatic discovery techniques or by using manual processes.

Overview of IMS environment definitions

The IMS Database Administration folder in the Navigation window of the console displays the IMS components that have been defined in a hierarchical representation (a tree).
From the sysplex level to the segment level, each object represents a component. You can interact with IMS components on the mainframe by selecting them from the tree in the console.

**Figure 18: IMS Database Administration folder in the Navigation window**

![Image of IMS Database Administration folder in the Navigation window]

**IMS objects on the tree**

The tree simplifies your interaction with the IMS objects in your environment by organizing the objects hierarchically.

Hierarchical organization also makes the process of defining parameters, such as thresholds, efficient. Parameters can be defined at the top level and can be inherited by all (or certain) child objects.

Each object has an associated pop-up menu that provides commands to add and configure child objects, to perform an action against an object, and to customize parameters for an object. For a comprehensive list of the objects and the commands that are available at each object level, see the online Help.

Table 6 on page 117 lists some objects that can be displayed in the IMS Database Administration folder.
Table 6: Objects in the IMS Database Administration folder

<table>
<thead>
<tr>
<th>Icon</th>
<th>Node name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>Host connection (connected)</td>
<td>Host connection and sysplex level</td>
</tr>
<tr>
<td>![Icon]</td>
<td>IMS Databases (for example)</td>
<td>Functional environment (such as the IMS Databases environment) in which one or more BMC products or components are installed and available for use</td>
</tr>
<tr>
<td>![Icon]</td>
<td>IMSPLEX</td>
<td>IMSPLEX (as identified by the RECON data sets)</td>
</tr>
<tr>
<td>![Icon]</td>
<td>IMS subsystem</td>
<td>IMS instance</td>
</tr>
<tr>
<td>![Icon]</td>
<td>DBRC Resources</td>
<td>Collection of DBRC resource objects, such as change accumulation groups, that are defined in the IMS RECON</td>
</tr>
<tr>
<td>![Icon]</td>
<td>CA Groups</td>
<td>Set of change accumulation (CA) groups that are defined in the IMS RECON</td>
</tr>
<tr>
<td>![Icon]</td>
<td>CA group</td>
<td>CA group that is defined in the IMS RECON</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Database group</td>
<td>Group of related databases</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Database</td>
<td>IMS full-function, HALDB, or Fast Path database</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Partition</td>
<td>HALDB partition</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Area</td>
<td>DEDB area</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Data set</td>
<td>IMS data set</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Segment</td>
<td>IMS segment</td>
</tr>
</tbody>
</table>

Choosing a method to define your IMS environment

To define your IMS database administration environment, you can use the Auto Configure wizard or define components manually. BMC recommends that you use the Auto Configure wizard.

The Auto Configure wizard guides you through the step-by-step process of defining an IMS database administration environment. When you make simple selections, the wizard automatically discovers the objects in the environment. When discovery is complete, the tree in the Navigation window is populated with the IMSPLEX, IMS subsystems, and database groups that you selected during the Auto Configure process. For more information about defining your IMS environment by using the Auto Configure wizard, see “Defining IMS components by using the Auto Configure wizard” on page 118.
Defining an IMS database administration environment manually requires more time and preparation than defining the environment by using the Auto Configure wizard. You must define an IMSPLEX, define an IMS subsystem, define database groups, register databases, and identify parameter definitions that you want to use to define each component. For more information about defining an environment manually, see “Defining IMS components manually” on page 123.

Defining IMS components by using the Auto Configure wizard

The Auto Configure wizard gathers information that is used to discover IMSPLEX configurations and to find and populate databases, ddnames, and data sets for all IMS subsystems.

You can also use the Auto Configure wizard to add existing database groups (that are defined in the RECON) to an environment.

When the Auto Configure process is complete, your IMS database administration environment is defined, the tree is displayed in the Navigation window, and all components in that environment are registered for use.

Typically, you run the Auto Configure wizard once for each host connection that you use. To take advantage of the automatic discovery process, you may also want to run the Auto Configure wizard any time that you add an IMSPLEX to the tree. When an IMS database administration environment is set up for a host connection, you can manage the environment directly from the tree.

Identifying information for the Auto Configure wizard

The Auto Configure wizard automatically populates most wizard pages so that you can select choices from a list.

Although you can specify values on some pages, all required fields are populated with default values that you can accept.

Table 7 on page 118 lists fields at IMS subsystem and IMSPLEX levels that the Auto Configure wizard populates. The table also identifies fields for which you might need to provide information during the Auto Configure process.

Table 7: Auto Configure fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Level</th>
<th>Source</th>
<th>User action</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMS Subsystem</td>
<td>IMS subsystem</td>
<td>z/OS</td>
<td>No action required</td>
</tr>
<tr>
<td>Field</td>
<td>Level</td>
<td>Source</td>
<td>User action</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------</td>
<td>--------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>IMS RESLIB</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>MODSTAT</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>Online Change Mode</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>ACBLIB</td>
<td>IMS subsystem</td>
<td>Proposed data set name based on information in the IMS control region</td>
<td>Accept the default value, or specify a new value</td>
</tr>
<tr>
<td>ACBLIBA</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>ACBLIBB</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>DBDLIB</td>
<td>IMS subsystem</td>
<td>Proposed data set name based on information in the IMS control region</td>
<td>Accept the default value, or specify a new value</td>
</tr>
<tr>
<td>DBDSRC</td>
<td>IMS subsystem</td>
<td>Not populated because the field is not available</td>
<td>(optional) Specify data set names</td>
</tr>
<tr>
<td>MDALIB</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>NEWDBD</td>
<td>IMS subsystem</td>
<td>Not populated because the field is not available</td>
<td>(optional) Specify data set names</td>
</tr>
<tr>
<td>PFXLIB</td>
<td>IMS subsystem</td>
<td>Proposed data set name based on information in the IMS control region</td>
<td>Accept the default value, or specify a new value</td>
</tr>
<tr>
<td>PFXLIBA</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>PFXLIBB</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>PSBLIB</td>
<td>IMS subsystem</td>
<td>Proposed data set name based on information in the IMS control region</td>
<td>Accept the default value, or specify a new value</td>
</tr>
<tr>
<td>PSBSRC</td>
<td>IMS subsystem</td>
<td>Not populated because the field is not available</td>
<td>(optional) Specify data set names</td>
</tr>
<tr>
<td>STEPLIB</td>
<td>IMS subsystem</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>Description</td>
<td>IMSPLEX</td>
<td>Populated by the wizard if (AutoConfigure) is appended to the description</td>
<td>(optional) Specify a different description</td>
</tr>
<tr>
<td>RECONx Data Set Name</td>
<td>IMSPLEX</td>
<td>IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>GSG Name</td>
<td>IMSPLEX</td>
<td>If specified, populated from IMS control region</td>
<td>No action required</td>
</tr>
<tr>
<td>RSE Name</td>
<td>IMSPLEX</td>
<td>If specified, populated from IMS control region</td>
<td>No action required</td>
</tr>
</tbody>
</table>
Starting the Auto Configure wizard

To start the Auto Configure wizard, right-click **IMS Databases** in the Navigation window and choose **Configuration => Auto Configure**. The Auto Configure wizard is displayed.

The Auto Configure wizard prompts you to make selections about the IMS database administration environment that you want to define. During the Auto Configure process, you can use the following buttons in the wizard:

- **Back** returns to the previous page.
- **Next** proceeds to the next page.
- **Cancel** exits the wizard without performing the task.

Selecting IMS RECON data sets

The Auto Configure wizard discovers all RECONs in your sysplex by examining IMS control regions across the sysplex. The wizard lists the RECONs on the Choose RECON Data Sets page. You may have to wait a few seconds for the list of RECONs to be populated.

Each row on the page represents a RECON that is defined for an IMSPLEX. The value under the **Configuration** column heading indicates whether the IMSPLEX has been defined in the IMS database administration environment in the console.

To select IMS RECON data sets

1. Select the RECONs that you want to use to build the IMSPLEX, and click **Next**.
2. *(optional)* To edit an IMSPLEX that has been defined, you must access the IMSPLEX object in the Navigation window and make the changes manually.

Selecting an IMS subsystem

The Auto Configure wizard discovers all active IMS subsystems that are associated with the IMSPLEX which was selected on the Choose RECON Data Sets page by examining your operating system. The wizard lists the subsystems on the Select IMS Subsystem page.

The value under the **Configuration** column heading indicates whether an IMS subsystem has been defined in the IMS database administration environment in the console. You can select only an IMS subsystem that has not been defined.

The Auto Configure wizard discovers an IMSPLEX only if it is active. The IMSPLEX can be on the same operating system image (LPAR) as the ADV server or on a
different operating system image if a connected ADV server is also running on that image.

To select an IMS subsystem

1 Select the IMS subsystem that you want to use to build the IMSPLEX, and click Next.

Note

To edit an IMSPLEX that has been defined in the IMS database administration environment in the console, you must access the IMSPLEX object in the Navigation window and make the changes manually. The names of the RECON data sets cannot be changed.

Defining library data sets

The Auto Configure wizard displays a series of pages that you can use to select or specify library data sets to associate with an IMS subsystem.

For most libraries, the wizard proposes a data set name that is based on information in the IMS control region.

You can select or specify data set names for the following libraries:

- ACBLIB
- DBDLIB
- PSBLIB
- ACBLIBA
- ACBLIBB
- PFXLIB
- PFXLIBA
- PFXLIBB
- MDALIB
Note

If you are using IMS Version 14.1 or later, you can select or specify IMS Catalog Directories in place of or in conjunction with the specified DBDLIBs and ACBLIBs. If you are performing an initial discovery on an IMS system and the IMS online environment has DIRECTORY=YES specified, Advisor builds a DD entry for CATBSDS in the IMS Subsystem section of Advisor.

You can also manually add the CATBSDS for systems that have already been built. It is the data set name used for the Directory BSDS used by IBM. Advisor then builds the tree of DBDs for the console by using the IMS Catalog Directory described in the BSDS to resolve relationships for DBDs, segments, and DSGs.

If you revert your IMS system to DIRECTORY=NO or you do not want to use the IMS Catalog Directory, set the CATBSDS of the IMS Environment section in the console to blanks.

To define library data sets

1. On each wizard page, select the proposed data set or specify a different data set.
2. Click Next.
   - If the proposed name is not the data set that you want to use, edit the name or delete the name and specify a different name.
3. (optional) To specify a data set name that is not populated in the list automatically, click Add or Find.
4. (optional) To change the order of data sets in the concatenation, select a data set name and click Move Up and Move Down.
   - For more information about adding library data set names, see the online Help.

Defining database groups

By examining the selected RECONs, the Auto Configure wizard discovers change accumulation groups and database data set groups that are defined in the RECONs. On the Define Database Groups page, the wizard lists the proposed names of candidate database groups to be created, based on definitions in the RECONs.

1. Select the check box under the Candidates column heading for each database group that you want to add.
2. (optional) Specify a description for each database group that you want to add.
   - If you do not specify a description, the group name is the default description. The description can be changed manually in the Database Group window at any time.
3 Click **Next**.

You can also define more database groups manually, based on DBD name, partition name, and area name. For more information about adding database groups manually, see **Adding groups manually from RECONs on page 127** and see the online Help.

**Reviewing Auto Configure information**

To complete the Auto Configure process, you can review Auto Configure information.

**To review Auto Configure information**

1 To complete the Auto Configure process, review the information that you specified and click **Finish**.

The Auto Configure wizard adds IMSPLEX, IMS subsystem, and database group definitions to the Navigation window. All databases in the IMSPLEX are registered for use.

**Defining IMS components manually**

To define IMS components manually, you add IMSPLEXs, IMS subsystems, and database groups to your IMS database administration environment one by one, outside of the Auto Configure process. When you define IMS components manually, you must provide all required information. Automatic discovery is not used to populate known fields or to suggest field values.

**Example**

Manual processes are appropriate when you want to edit library data sets that are associated with an IMS subsystem, to add a database group, or to edit database group criteria.

**Defining IMSPLEXes manually**

When you define an IMSPLEX manually, you define parameters such as RECON data sets and you create an IMSPLEX object on the tree to which you will add any IMS subsystems and database groups.
**Before you begin**

You must know the parameter values that you want to specify, including RECON data set names.

To define an IMSPLEX name that is used by SAF, you must also have the required security access. For more information about security access, see the *Database Products for IMS Customization Guide*.

**To define an IMSPLEX manually**

1. In the Navigation window, right-click the **IMS Databases** object that you want to contain the IMSPLEX and choose **Configuration => Add IMSPLEX**.

   The Add IMSPLEX window (Figure 19 on page 124) is displayed.

   **Figure 19: Add IMSPLEX window**

   ![Add IMSPLEX window](image)

2. Specify the required parameters, and click **Save**.

3. *(optional)* To search for data sets, click **Find**.

   For more information about each parameter, including descriptions and valid values, see the online Help.

**Defining IMS subsystems manually**

IMS subsystems are defined from the IMSPLEX object, so you must have defined an IMSPLEX before you can define an IMS subsystem. Before defining the IMS subsystem, you must know the parameter values that you want to specify, including the data set names to associate with each ddname.
To define an IMS subsystem manually

1. In the Navigation window, right-click the IMSPLEX where you want to define the IMS subsystem and choose Configuration => Add IMS Subsystem.

   The Add IMS Subsystem window (Figure 20 on page 125) is displayed.

Figure 20: Add IMS Subsystem window

2. Specify the required parameters.

   For more information about each parameter, including descriptions and valid values, see the online Help.

3. Add IMS data set names.

4. (optional) To search for data sets, click Find.

5. (optional) To change the order of data sets in the concatenation, select a data set name and click Move Up and Move Down.

6. Click Save.

Defining database groups manually

Adding database groups manually is required when you want to define a group that is based on DBD names, partition names, or area names.
Although you can add database groups that are based on the change accumulation (CA) name and the database data set (DBDS) name during the Auto Configure process, you can also manually add groups that are based on those criteria. To define a database manually, see “Adding groups manually from RECONs” on page 127.

Working with database groups

Advisor work with groups of related databases to simplify database maintenance tasks and to make your actions efficient and effective.

Note

Recovery Manager (RMGR) groups, which are defined and used by the RMGR functions and utilities of the Backup and Recovery Solution for IMS product and the RECOVERY MANAGER for IMS product, are not used in the console.

Overview of groups and database ownership

A database group associates related databases, partitions, or areas into a single unit so that actions can be initiated against every item in the group.

You can create a database group that is based on one (or a combination) of the following criteria:

- DBD name
- Change accumulation group name
- Database data set group name

You can qualify these criteria by specifying a partition name for HALDBs or an area name for DEDBs.

You can create database groups during the Auto Configure process when you define an IMS database administration environment. You can also create database groups manually.

A single database can belong to multiple database groups. Depending on the criteria that you specify when you create your database groups, a single database may meet the criteria for several groups. However, only one group can own the database. If parameter changes are made to a group, those changes are not inherited by databases that are owned by a different group.

Database ownership ensures the integrity of parameters that are defined for all database objects and for all objects that are subordinate to databases (partitions, areas, and data sets). Protected parameters include analysis options (such as
thresholds and goals) and site values (such as job control information). To customize a parameter for a database that belongs to more than one database group, you must select the database object that is displayed under the owning database group on the tree.

Adding groups during the Auto Configure process

Database groups that are based on CA group name or DBDS group name criteria can be added during the Auto Configure process.

The Auto Configure wizard populates the Define Database Groups page with all database groups (based on CA and DBDS names) that are defined to DBRC. For more information, see “Starting the Auto Configure wizard” on page 120.

Adding groups manually from RECONs

To define a group that is based on DBD name, partition name, or area name, you must add the group manually. Although you can add database groups that are based on CA and DBDS names during the Auto Configure process, you can also manually add groups that are based on those criteria.

To add a database group manually

1 In the Navigation window, right-click the IMSPLEX that you want to contain the database group and choose Configuration => Add Database Group.
The Add Database Group window is displayed (Figure 21 on page 128). Use this window to define a database group.

**Figure 21: Add Database Group window**

2 Specify a database group name and description.

For more information about each parameter, including descriptions and valid values, see the online Help.

3 Click Add Criteria.
The Database Group Criteria dialog box (Figure 22 on page 129) is displayed.

**Figure 22: Database Group Criteria dialog box**

![Database Group Criteria dialog box](image)

4 Specify criteria for the database group, and click **OK**.

The dialog box is closed, and the Add Database Group window is displayed. The group criteria that you just specified are displayed in the window. To add more group criteria for the database, repeat Step 3 on page 128 and Step 4 on page 129.

For more information about database groups, see the online Help.

**Managing groups by using the Assigned Databases report**

You can manage database groups from the Assigned Databases report. The report lists all databases that are assigned to the selected database group.

**To manage groups by using the Assigned Databases report**

1 In the Navigation window, right-click a database group and choose **Reports => Assigned Databases**.
The Assigned Databases report (Figure 23 on page 130) is displayed. The report lists all databases that are assigned to the database group.

Figure 23: Assigned Databases report

![Assigned Databases report](image)

2 Perform one of the following actions:

- To add a database in the list to an existing database group, right-click the database row and choose **Add to Existing Group**.

  The Add to Existing Group dialog box (Figure 24 on page 130) is displayed. The dialog box lists database groups in the IMSPLEX. Select the group to contain the database, indicate whether to include the indexes and logically related objects of the database, and click **OK**.

  Figure 24: Add to Existing Group dialog box

  ![Add to Existing Group dialog box](image)

- To add a database to a new group, right-click the database row and choose **Add to Group (Create New Group)**.
The Create Database Group dialog box (Figure 25 on page 131) is displayed. Use this dialog box to define a database group.

**Figure 25: Create Database Group dialog box**

Managing groups by using the Unassigned Databases report

You can manage database groups from the Unassigned Databases report. The report lists all databases in the IMSPLEX that are not assigned to a database group but are registered in the IMS database administration environment. If a database is not assigned to a database group, it is not owned by a database group.

**To manage groups by using the Unassigned Databases report**

1. In the Navigation window, right-click an IMSPLEX object and choose **Reports => Unassigned Databases**.
The Unassigned Databases report (Figure 26 on page 132) is displayed. The report lists all databases that are not owned by a database group.

Figure 26: Unassigned Databases report

2 Add the database to an existing database group or a new database group (Step 2 on page 130).

Deleting groups

You can delete database groups. If you delete a database group, databases that are members of other groups are not deleted or removed from those groups. If a database is owned by the deleted group, the database becomes unowned.

To delete a database group

1 In the Navigation window, right-click the database group that you want to delete and choose Configuration => Delete Database Group.

2 In the Database Group window, click Delete.

The database group is deleted.

Renaming groups

You can rename database groups.
To rename a database group, you must delete the database group and redefine the group. You might want to review the database group criteria before deleting the group so that you can recreate it quickly. Multiple group criteria might be defined for the group. When you delete a database group, databases that are members of other groups are not removed or deleted from those groups. If a database is owned by the deleted group, the database becomes unowned.

To rename a database group

1. In the Navigation window, right-click the database group that you want to rename and choose Configuration => Maintain Database Group.

   A table summarizing all group criteria that are specified for the group is displayed in the Database Group window. Note the criteria on which the database group is based.

2. Delete the group, as explained in “Deleting groups” on page 132."

3. Redefine the database group by using the new name. To define a database group, see “Adding groups manually from RECONs” on page 127.

Managing database ownership

Although a database might belong to multiple database groups, each database is owned by only one database group.

To customize a parameter for a database that occurs in more than one database group, you must select the database object that is displayed under the database group which owns the database.

Example

Database db1 was first added to database group dbg1. The result is that database group dbg1 owns database db1. Creation of more database groups resulted in database db1 being added to database group dbg2 and database group dbg3. When database group dbg3 was created, ownership of all databases in database group dbg3 was transferred to database group dbg3. Now database db1 is owned by database group dbg3.

To customize any parameters for database db1, you must select the database db1 that is in database group dbg3. To customize parameters for any object that is subordinate to database db1, you must select the subordinate object under the database db1 object that is displayed under database group dbg3.

Database ownership is established automatically only when you create database groups during the Auto Configure process. When you create a database group manually or manage groups by using the Assigned Databases report or the
Unassigned Databases report, you must also assign or transfer database ownership manually.

**Viewing database ownership information**

BMC recommends that you view database ownership information for a database by using the Manage Database Ownership window.

To view database ownership information for a database

1. In the Navigation window, right-click the database group that contains the databases for which you want to view ownership and choose **Configuration => Manage Database Ownership**.

The Manage Database Ownership window (Figure 27 on page 134) is displayed. You can use this window to view, assign, and transfer database ownership.

**Figure 27: Manage Database Ownership window**

Databases that are listed in the Manage Database Ownership window belong to the selected database group. The name and description of the database group are displayed at the top of the window.

If a database is owned by the selected group, a check mark is displayed in the **Owned** column. If a database is not owned by the selected database group, the name of the database group that owns the database is displayed in the **Owner** column.
A quick way to view database ownership information for an object is by displaying object "flyover." To display a flyover, position the cursor over an object in the Navigation window. The name of the database group that owns the database is displayed in the **Owning DB Group** field.

**Assigning or transferring database ownership**

BMC recommends assigning or transferring database ownership by using the Manage Database Ownership window.

*Note*

Databases that are not assigned to any group are considered "orphans." Orphan databases are not displayed in the Navigation window but remain registered in the IMS database administration environment.

**To assign or transfer database ownership**

1. In the Navigation window, right-click the database group that contains the databases for which you want to manage ownership and choose *Configuration* => *Manage Database Ownership*.

2. Perform one or more of the following actions:

   - To transfer ownership of a database in the selected database group to the selected database group, select the check box in the **Owned** column.

   - To transfer ownership of all databases in the selected database group to the selected database group, click **Select All**.

   - To remove a database from the selected database group, clear the check box in the **Owned** column.

     The database becomes unowned.

   - To remove all databases from the selected database group, click **Unselect All**.

     The databases become unowned.

3. Click **Save**.

For more information about database ownership, including descriptions of all column headings in the Manage Database Ownership window, see the online Help.
Searching for databases

You can use the database search feature to locate a database (or several related databases) across an IMSPLEX.

Overview of the search feature

With the database search feature, you can perform a search that is based on a DBD name.

You can qualify the search by specifying a partition name or an area name. Search results include all registered and unregistered databases in the IMSPLEX.

Example

You might want to create a database group that is based on the DBD name. You could use the database search feature to preview the list of databases that would be included in the group.

For another example, an exception might be reported for a database that is not assigned to a database group. Although you can access information about the database directly from the Exception List, you might want to use the search feature to display the database object in the Navigation window. (A database that is not assigned to a database group is not displayed in the Navigation window.)

Specifying search criteria

You can perform a search for databases across an IMSPLEX.

To search for databases across an IMSPLEX

1 In the Navigation window, right-click an IMSPLEX and choose Search Databases.
The Search Databases dialog box (Figure 28 on page 137) is displayed. Use this dialog box to specify the criteria to use when searching for databases.

**Figure 28: Search Databases dialog box**

![Search Databases dialog box](image)

2 Specify search criteria, and click **Search**.

For more information about search criteria, see the online Help.

Search results are displayed in the Navigation window (Figure 29 on page 137) as a temporary list of databases. You can access the pop-up menu for any registered database that is displayed in the search results. Search results persist until you exit the console or you delete them.

**Figure 29: Database search results**

![Database search results](image)

### Deleting search results

You can delete search results in the Database Search Results folder:

1 In the Navigation window, right-click the Database Search Results folder and choose **Delete Database Search Result**.
Using Database Advisor

This chapter contains information about working with Database Advisor exceptions and solutions.

Overview of Database Advisor processes

Database Advisor and other processes work together to collect data about your databases and environment, to analyze the data that is collected, and to recommend the best solution to correct reported problems.

Note

This section applies only to Database Advisor. Recovery Advisor processes differ from Database Advisor processes in important ways. For more information, see Overview of Recovery Advisor processes on page 173.

Collect, Analyze, and Solution processes are controlled by user-defined parameters and scheduled functions. BMC populates suggested defaults for most parameters, so you can begin using Database Advisor as soon as possible. The schedules that control Collect and Analyze processes for full-function and HALDB databases are activated by default. You must activate Collect and Analyze processes for Fast Path DEDBs explicitly (by selecting the RECON object in the tree, selecting Configure => Schedules, and editing the appropriate FP schedule).

The populated defaults make it easy for inexperienced DBAs to start using Database Advisor while learning more about sound data management practices. Using customization options, experienced DBAs can fine-tune parameters and schedules that control Database Advisor processes.

You can customize most parameters and schedules at most levels in the object tree, but BMC recommends that you start by specifying values at the host level. All subordinate levels inherit the values. You can customize subordinate levels at any time.
Collect process

The Collect process fulfills the Gather process of the Database Advisor database maintenance strategy.

Database Advisor gathers data about your databases and your environment for use in predictive analysis (as explained in “Advisor database maintenance cycle” on page 35).

Data that is collected about your databases includes key elements, such as data set size and I/O probability, about the state of each database. Database Advisor uses this data to determine space and performance conditions for a database. Data that is collected about your environment includes information about recent events that may have affected your databases. Database Advisor uses this data to determine whether the end-of-day process should be executed.

Collecting data about your databases

Database Advisor uses the following controls to automate the collection of data about your databases:

- FF COLLECT function schedule (for full-function databases and HALDBs)
- FF COLLECT HOURLY function schedule (for full-function databases and HALDBs)
- FP COLLECT function schedule (for Fast Path databases)
- FP SPACE COLLECT function schedule (for Fast Path databases)
- Data Collection Interval parameter
- Activate Fast Path Space Data Collection parameter

The Data Collection Interval specifies the frequency of collecting data for a database. The suggested default that is populated at the host level is 28 days.

Database Advisor provides a separate data collection schedule to collect IOVF and SDEP data for Fast Path databases. The FP SPACE COLLECT function can be scheduled to collect Fast Path space data automatically in hourly intervals. Because of potential resource usage requirements, by default the schedule is not activated.

When the selected COLLECT function is executed, Database Advisor examines the collection interval parameter value for each database. If the value indicates that it is time to collect data for the database, Database Advisor collects current data for the database.
Example

If the Data Collection Interval value for database db1 is 30 and the COLLECT function determines that data has not been collected for database db1 in 30 or more days, current state data is collected and stored. If the COLLECT function determines that data was collected for database db1 29 or fewer days ago, data is not collected.

Collecting data about your environment

One of the great advantages of using BMC utilities with Database Advisor is that the utilities communicate with Database Advisor to initiate data collection and subsequent analysis.

End-of-day processing is the mechanism by which Database Advisor tracks events that are controlled by non-BMC utilities.

Database Advisor uses the ENDOFDAY function schedule to automate collection of data about your environment. This function analyzes events of the day to determine which tasks must be performed. Has the state of a database changed? If so, data collection must be performed. Have base values for a database changed? If so, base values are updated for the database.

A full-function database is included for end-of-day processing when POINTER CHECKER PLUS is not executed after the following events:

- A database load utility is executed.
- An IDCAMS REPRO is performed on a database data set.
- A database is recovered.

Setting the frequency of the Collect process

BMC recommends that you use suggested defaults for the schedules and related parameters that control the Collect process.

By default, FF COLLECT, FP COLLECT, and ENDOFDAY functions are scheduled to execute daily. The Data Collection Interval default is 28 days. The FF COLLECT HOURLY function can be scheduled at intervals that are smaller than one day.

You can customize schedules or related parameters as necessary. For example, you might want to decrease the frequency of a COLLECT function or increase the Data Collection Interval for a database group that you know experiences little processing.

COLLECT functions are also initiated by the following events:

- You can issue a request to execute FF COLLECT and FP COLLECT functions on demand.
COLLECT functions might be executed automatically, based on information that was gathered during end-of-day processing (ENDOFDAY function). For example, if end-of-day processing determines that base values must be updated, COLLECT functions are executed to collect the data.

COLLECT functions are executed automatically when any of the following BMC products, utilities, and functions are executed:

- MAXM Reorg/Online for IMS
- MAXM Reorg/EP for IMS
- MAXM Reorg/EP for IMS with Online/Defrag Feature
- MAXM Reorg/EP Express for IMS
- UNLOAD PLUS/EP for IMS
- LOADPLUS/EP for IMS
- SECONDARY INDEX UTILITY/EP
- FAST REORG FACILITY/EP
- CHANGE RECORDING FACILITY
- POINTER CHECKER PLUS®
- BMC Fast Path/EP Analyzer (standalone or under other Fast Path functions)

When a database is reloaded by using the IBM reorganization utility, Database Advisor recognizes that event and executes immediate data collection or schedules data collection for the end of the day, depending on user-controlled parameters.

**Storing collected data**

Collected data is stored in self-managing repositories. For more information about repository backups, purges, and expansions, see the *Database Products for IMS Customization Guide*.

**Analyze process**

The Analyze process fulfills the Analyze process of the Database Advisor database maintenance strategy.

The Analyze process fulfills the Analyze process of the Database Advisor database maintenance strategy (as explained in “Advisor database maintenance cycle” on page 44).

Database Advisor analyzes collected information about your databases to find and forecast problems. Existing and potential problems are reported to the Exception List. You can specify that Database Advisor notify you when an exception is reported to the Exception List.
Basic analysis

The Analyze process uses the following information to determine whether thresholds have been met or exceeded:

- Base values
- Current state values
- Threshold values

Base values reflect the optimal state of a database. Current state values are the most current data that is collected about a database. Thresholds are acceptable deviations from base values. During analysis, thresholds are compared to current state values to calculate the rate at which a threshold is being approached or to determine whether the threshold has been met or exceeded.

Base values

Base values are stored for database, partition, area, and data set objects. During analysis, Database Advisor compares base values with current state information to assess changes to an object. By monitoring the changes, Database Advisor identifies threshold violations.

Base values are initially set differently for full-function databases and Fast Path DEDBs.

Full-function database base values

When a full-function database is registered with Database Advisor, base values are initially set by one of the following methods:

- Data is collected from subsequent execution of POINTER CHECKER PLUS.
- Data is collected from subsequent execution of COLLECT functions (scheduled or on demand).
- Database-level base values and partition-level base values can also be set when data is collected from subsequent execution of a BMC reorganization utility.

Base values are updated automatically after execution of any of the following BMC utilities:

- LOADPLUS/EP for IMS
- SECONDARY INDEX UTILITY/EP
- Single-step DBR Reorg
- Single-step online Reorg
When LOADPLUS/EP for IMS or SECONDARY INDEX UTILITY/EP is executed, base values for databases and partitions are set and base values for data sets are cleared. Data set base values in these cases are reset by one of the following methods:

- Data is collected from subsequent execution of POINTER CHECKER PLUS.
- Data is collected from subsequent execution of the ENDOFDAY function.

If a database is reorganized with an IBM reorganization utility, Database Advisor resets the base values for the database with the next ENDOFDAY run.

**Fast Path DEDB base values**

The base values for a Fast Path DEDB are determined differently from base values for a full-function database.

When Database Advisor posts a current set of statistics, it searches the repository for statistics records that were posted previously. Database Advisor tries to find a corresponding record type (FULL or SPACE) for the current set of statistics to be posted. If Database Advisor cannot find a corresponding record type, it locates the record that represents the best match. If Database Advisor finds a record-type match, it selects the record as the base and the search terminates. If Database Advisor does not find a record-type match, it keeps track of the best match found; older records at the same match level are considered to be better than more recent ones.

When a BMC Fast Path/EP Analyzer RELOAD or CHANGE command process with the ANALYZE function (POINTER_VALIDATION keyword) is executed, the new repository record becomes the base and all previous records are ignored.

**Current state values**

Current state values reflect the current state of a database. This data is updated with every new data collection event. The current state values that are used during analysis are the most current information that has been collected about the database, partition, or data set.

You can view current state values from the following reports:

- Database State report
- HALDB State report
- Partition State report
- Data Set State report
- AREA State report

For more information about current state values, see “Evaluating the current state of your IMS data” on page 430.
Thresholds

Thresholds are rules that establish an acceptable deviation from a base value.

Database Advisor uses thresholds to monitor the following space and performance conditions of a database:

- Internal free space
- **z/OS data set allocation for full-function databases:**
  - Extents
  - Data set size
  - Candidate volumes
  - CA/CI splits
  - SMS support
- Probability of I/O
- Fragmentation

When a threshold is violated, an entry is added to the Exception List. For example, if the threshold for CA splits is 20 percent, Database Advisor reports an exception to the Exception List when the percentage of CA splits meets or exceeds 20 percent.

Database Advisor goes beyond just reporting threshold violations. Database Advisor tracks the rate at which thresholds are being approached and adds an entry to the Exception to warn you about potential problems.

You can define a parameter to tell Database Advisor how far in advance that you want to receive warnings. For example, you can tell Database Advisor to warn you 30 days before a problem is expected to occur.

**Two-phase analysis**

Database Advisor uses a two-phase analysis process. Analysis is automatically executed any time Database Advisor receives data about a registered object, namely when collection processing updates data in the repositories.

In phase one, current state values are compared to the corresponding thresholds and exceptions are reported to the Exception List. If phase one determines that the probability of I/O is increasing at a rate that could negatively affect database performance, an exception is added to the Exception List and Database Advisor starts phase two of analysis.
In phase two, Database Advisor reads the entire database to determine the percentage of records in the database that can be improved and the degree to which the records can be improved.

If Database Advisor determines that the level of database fragmentation cannot be improved significantly, the exception that was reported to the Exception List during phase one (that the probability of I/O growth was too high) is removed from the Exception List. To avoid a repeat of this situation, Database Advisor dynamically increases the value of the threshold that initiated the exception.

If Database Advisor determines that the level of database fragmentation requires your attention, the exception that was reported to the Exception List during phase one (that the probability of I/O growth was too high) remains on the Exception List. Database Advisor adds another exception to the Exception List to reflect that too many database records are fragmented.

**Analysis reporting**

When Database Advisor discovers a threshold violation during analysis, an exception is reported to the Exception List.

If E-mail/WTO parameters are enabled, a notification message is generated when an exception is added to the Exception List or when an exception changes status (for example, from warning to critical). Messages for individual exceptions can be sent immediately or consolidated into one message that is sent on a scheduled basis.

**Ensuring that analysis has occurred**

You can perform a periodic audit of your databases to verify that data has been collected and analyzed as expected. The audit can report a potential problem as an exception in the Exception List or in a batch report.

For more information, see “Auditing databases to ensure analysis” on page 170.

**Solution process**

When you select an object in the Exception List window and click Solutions, the Solutions window is displayed.

This window lists the recommended solution to correct detected exceptions, and you can request the display of solutions to correct selected exceptions. A solution is associated with a script that is used to customize skeleton JCL for executing a function or utility (such as a database reorganization). When the customized JCL is generated and executed, the problem that is indicated by the exception is corrected.
The solution process fulfills the execute portion of the strategy that is explained in “Advisor database maintenance cycle” on page 35. The process recommends the best solutions to correct problems that are reported by the Database Advisor analysis process. You can correct all (or certain) exceptions by using the recommended solutions or any available solutions.

**Note**

The processes that Recovery Advisor uses to recommend solutions and generate JCL also apply to Database Advisor. If Recovery Advisor and Database Advisor report exceptions for the same object, the solution process chooses the best solutions to correct exceptions.

When you select a solution, the Generate JCL function generates JCL that is optimized for your environment. When the generated JCL is executed, the problems that are indicated by the exceptions are corrected.

Parameters that control solution processing include goal parameters, job control parameters, and parameters for specific functions.

For more information about using solutions and generating JCL, see “Working with solutions, JCL generation, and viewing jobs” on page 323.

---

### Using Database Advisor to manage Reorg thresholds

You can use Database Advisor to manage Reorg thresholds.

1. **Set parameter values in the Maintain Parameters window.**

   To configure Database Advisor for Reorg thresholds management, you establish parameter values in many parameter categories of the Maintain Parameters window. For more information, see “Establishing Reorg threshold parameters” on page 148.

2. **Submit Database Advisor for execution by using one of the following methods:**

   - You can schedule Database Advisor to execute at predetermined times as described in “Customizing schedules” on page 279.
   - You can request execution of Database Advisor on demand as described in “Requesting Database Advisor processes on demand” on page 148.

3. **Recognize that a Reorg threshold exception has been detected.**
You can check the Exception List manually to look for reported Reorg threshold exceptions, or you can receive automated notification that a Reorg threshold exception has been detected. For more information, see “Recognizing Reorg threshold exceptions” on page 150.

4 Determine how to address the exception.

You can choose any of the following actions:

- From the Exception List, select a database object for which one or more exceptions were reported and click Solutions. Depending on the type of exceptions that were reported, Advisor can recommend a variety of solutions. For more information, see “Using solutions from the Exception List” on page 333.

- Analyze the exception by performing research. For more information, see “Analyzing Database Advisor exceptions” on page 162.

- If the exception is in Warning status, you can use the Exception Forecast report to determine when the status is projected to change. For more information, see “Using the Exception Forecast report” on page 168.

Establishing Reorg threshold parameters

The Database Advisor analysis process compares current statistics to established thresholds to find Reorg exceptions, which indicate problems that can have an adverse affect on database availability and performance.

You can control the detection of Reorg threshold exceptions by establishing Reorg thresholds in the Maintain Parameters window. You can set Reorg thresholds at host, IMSPLEX, and database group, database, and database data set levels. For more information, see “Reorg Thresholds parameters” on page 230.

For general information about working with parameters, see “Customizing parameters” on page 226.

Requesting Database Advisor processes on demand

You can initiate the following Database Advisor processes on demand:

- Collect process (FF COLLECT and FP COLLECT)
- Analyze process
- Database Registration Synchronization process (DBSYNCH)

You might want to request the Collect process or the Analyze process on demand to test customized parameter values or to determine the effects of resetting base values.

**Note**

These processes do not apply to Recovery Advisor. The RECOVERY ADVISOR function is executed according to the established schedule.

---

**Requesting data collection on demand**

In addition to scheduling automatic data collection, you can also request data collection on demand.

To request data collection on demand, in the Navigation window right-click the database or partition for which you want to request data collection and choose Request Data Collection.

**Requesting analysis on demand**

In addition to the automatic analysis that follows data collection, you can request an analysis for a selected database or partition.

You can also request analysis for all database objects in a selected IMSPLEX (identified in the Navigation window by the RECON name).

To request analysis of database objects on demand, in the Navigation window right-click the RECON, database, or partition object for which you want to request analysis and choose Request Analysis.

This request does not affect the RECON data sets; they are analyzed during the Recovery Collect scheduled process.

**Requesting database registration synchronization on demand**

In addition to scheduling automatic synchronization of database registration, you can also synchronize database registration on demand.

To request synchronization on demand, in the Navigation window right-click the database or partition for which you want to request synchronization and choose Synchronize.
Recognizing Reorg threshold exceptions

When the Database Advisor Analyze process is executed, it compares established Reorg threshold values with current values to detect and report Reorg threshold exceptions.

This section explains Reorg threshold exceptions. For general information about exceptions, see “Handling exceptions” on page 295.

AREA DOVF UTILIZATION TOO HIGH exception

When the percentage of allocated dependent overflow (DOVF) control intervals for a DEDB meets or exceeds the value of the DOVF Space Utilization % threshold parameter, this exception entry is critical (red).

The statistics collector uses the total number of DOVF blocks in the area, and the number of allocated or used DOVF blocks. The percent of DOVF allocation is calculated and compared or projected against the threshold.

The solution for this exception is to extend the number of DOVF blocks in the area.

AREA FRAGMENTATION TOO HIGH exception

When the fragmentation or disorganization factor for a DEDB area meets or exceeds the threshold, this exception entry is critical (red).

This value is based on the following assumptions:

- Only one free space element will be available in a block. Excessive numbers of free space elements within a block indicate disorganization.

- All RBA pointers within a block, except one, will fall in the same block. Excessive numbers of pointers that point outside of the block indicate disorganization.
AREA I/O PROBABILITY TOO HIGH exception

When the number of I/O operations that are required for reading a database record increases by the threshold percentage, this exception entry is critical (red).

The statistics collector determines the number of I/O operations that are required for reading an average database record from the area, compares this information to the historical base point, and is projected against the threshold. High I/O probability is a measurement of the disorganization of the area.

AREA IOVF SPACE LOW exception

When the percentage of allocated IOVF control intervals for a DEDB meets or exceeds the threshold, this exception entry is critical (red).

The statistics collector uses the total number of IOVF blocks in the area and the number of allocated or used IOVF blocks. The percentage of IOVF allocation is calculated and is compared to or projected against the threshold. The fragmentation level of the area also helps determine the solution to the exception.

AREA RAA UTILIZATION TOO HIGH exception

When the percentage of allocated root addressable area (RAA) control intervals for a DEDB meets or exceeds the value of the RAA Space Utilization % threshold parameter, this exception entry is critical (red).

The statistics collector uses the total number of RAA blocks in the area, and the number of allocated or used RAA blocks. The percent of RAA allocation is calculated and compared or projected against the threshold.

The solution for this exception is to extend the number of RAA blocks in the area.

AREA SDEP FREE SPACE TOO LOW exception

When the percentage of allocated SDEP control intervals for a DEDB meets or exceeds the threshold, this exception entry is critical (red).

The statistics collector uses the total number of SDEP blocks in the area and the number of allocated or used SDEP blocks. The percentage of SDEP allocation is calculated and is compared to or projected against the threshold.
AREA USED PERCENT EXCEEDED exception

When the percentage of a DEDB area that is used meets or exceeds the threshold, this exception entry is dead (black).

The solution to this exception is to review the ROOT and UOW parameters in the DBD.

FSE ERROR exception

When data collection detects a free space element error in the database, Database Advisor reports the FSE ERROR exception to the Exception List.

HDAM RAA DEFINITION TOO LARGE exception

When the percentage of blocks in the RAA meets or exceeds the value that is specified for the RAA Larger by % threshold, this exception entry is critical (red). This exception never has a warning (yellow) status or a dead (black) status.

The RAA Larger by % threshold is associated with this exception.

HDAM RAA DEFINITION TOO SMALL exception

When the percentage of blocks in the RAA meets or falls below the value that is specified for the RAA Smaller by % threshold, this exception entry is critical (red). This exception never has a warning (yellow) status or a dead (black) status.

The RAA Smaller by % threshold is associated with this exception.

HIDAM PRIMARY INDEX ERROR exception

Data collection hash checking has detected the hash value for the HIDAM primary index pointers does not match the hash value for the HIDAM primary database root segment relative byte addresses.

The solution to this exception is to review the ROOT and UOW parameters in the DBD.
INCORRECT CI SIZE exception

When the size of the control interval is different from the established threshold value, this exception entry is dead (black). This exception indicates that the object was defined in a way that violates your standards.

There is no solution to this exception.

INSUFF ALLOC FREE SPACE (LS) exception

When the calculated percentage of allocated free space for the database falls below the value of the Alloc Freespace(LS) % threshold parameter, this exception entry is critical (red).

The current value is calculated by subtracting the high used relative byte address from the high allocated relative byte address and adding the freespace entities that can hold the largest segment size.

INSUFF TOTAL FREE SPACE (LS) exception

When the percentage of free space for the database, relative to the theoretical maximum amount of free space, falls below the value of the Total Free space(LS) % threshold parameter, the exception is critical (red). If the percentage of free space reaches the theoretical maximum, the exception entry is dead (black).

The Total Free Space(LS) % threshold incorporates extents in addition to the free space used for the Allocated Freespace(LS) % threshold.

INSUFFICIENT ALLOCATED FREE SPACE exception

When the calculated percentage of allocated free space for the database falls below the value of the Alloc Freespace % threshold parameter, this exception entry is critical (red).

The current value is calculated by subtracting the high-used relative byte address (RBA) from the high-allocated RBA and adding the results to the current amount of free space. Available extents are not included in the calculation of free space.
INSUFFICIENT FREE BLOCKS exception

When a data set has reached its last candidate volume, this exception entry is critical (red). The exception indicates that the data set is nearly out of growth potential and does not have much distributed free space for records (based on the IMS bitmap).

All potential future extents on the current volume are assumed to be available. All current and future available blocks are calculated. A percentage of available blocks divided by total data set blocks is calculated. If the calculated percentage is less than the threshold, the exception entry is critical. If available blocks reach zero, the exception entry is dead (black) because no new records can be inserted into the data set.

Example
A data set resides on five volumes and has no more candidates. Ten more extents are assumed to reside on the current volume, which is equivalent to 250,000 blocks. The bitmap shows that 50,000 blocks can hold new records. When the volume can take no more extents, the data set will contain 2 million blocks. The current value for Free Blocks % is 15 (50,000 bitmap blocks plus 250,000 future blocks divided by 2 million blocks equals 15 percent). If the calculated percentage is less than the threshold, the exception is critical.

Example
A data set has 180,000 blocks and has no secondary extents defined. According to the bitmap, the data set has 100,000 free blocks. The threshold is set to 10 percent (18,000 blocks). The bitmap changes as data is added and blocks become fragmented. When only 18,000 blocks are free, the exception entry is critical.

Note
An index has no distributed free space or bitmap. All future potential extents are considered free blocks. All allocated but unused blocks are free. Future extents and unused blocks are compared against the threshold to determine the status of an exception entry.

The following thresholds are associated with this exception:

- OSAM Data Set Size
- VSAM Data Set Size

INSUFFICIENT FREE SPACE IN RAA exception

When the percentage of free space in the RAA falls below the value that is specified for the RAA Free Space Minimum threshold, this exception entry is critical (red).
You can stop Database Advisor from reporting this exception by specifying zero for the RAA Free Space Minimum threshold.

The RAA Free Space Minimum threshold is associated with this exception.

**INSUFFICIENT SPACE exception**

When the size of a data set meets or exceeds the threshold, this exception entry is critical (red). If the data set size reaches the IMS maximum for the data set access method, the exception entry is dead (black).

**INSUFFICIENT TOTAL FREE SPACE exception**

When the percentage of free space for the database, relative to the theoretical maximum amount of free space, falls below the value of the Total Freespace % threshold parameter, this exception entry is critical (red).

If the percentage of free space reaches the theoretical maximum, the exception entry is dead (black).

**I/O PROBABILITY TOO HIGH exception**

This exception entry is created when the I/O Probability Growth % threshold is met or exceeded. The exception entry is added to the Exception List, and phase two of analysis determines whether the exception remains on the Exception List.

The Probability of I/O base value and current state value are calculated by using sequential forward pointer types (RAP, PTF, PCF, HF) in a database. The Probability of I/O base value is stored after a reorganization. The I/O Probability Growth % threshold is added to the base value to calculate a value that will trigger an exception. When the Probability of I/O current state value exceeds the trigger value, the exception entry is critical (red).

For most databases, a more detailed analysis (phase two) is performed by using Improvement % and Fragmentation % thresholds to determine whether the exception should remain on the Exception List.

**Note**

To prevent unnecessary reorganization, analysis uses a minimum value of 10 percent for the Probability of I/O base value.
Example
If the Probability of I/O base value for a database is 10 percent after reload and increases to 30 percent over time, the I/O probability growth is 200 percent. If the I/O Probability Growth % threshold is 200 or less, phase two of analysis checks Improvement % and Fragmentation % thresholds. If both thresholds are met or exceeded, the exception remains on the Exception List. If only one or neither of the thresholds is met or exceeded, the exception entry is removed from the Exception List and the I/O Probability Growth % threshold is increased dynamically.

Example
If the Probability of I/O base value for a database is 2 percent after reload and the I/O Probability Growth % threshold is 50 percent, analysis overrides the base value of two with a base value of 10. When the current state Probability of I/O reaches the trigger value of 15 percent (50 percent growth rate added to 10 percent base value), the exception entry is critical.

The following thresholds are associated with this exception:

- I/O Probability Growth %
- Improvement %
- Fragmentation %

MAXIMUM OSAM ALLOC EXCEEDED exception

The percentage of allocated cylinders for an OSAM database data set has exceeded the value specified in the Max Allocation Exceeded (Cyls) threshold for OSAM data sets.

MAXIMUM VSAM ALLOC EXCEEDED exception

The percentage of allocated cylinders for a VSAM database data set has exceeded the value specified in the Max Allocation Exceeded (Cyls) threshold for VSAM data sets.

POINTER ERROR exception

When data collection hash checking detects a pointer error in the database, Database Advisor reports the POINTER ERROR exception to the Exception List.
PRILOG SIZE EXCEEDS THRESHOLD exception

Recovery Advisor reports the PRILOG SIZE EXCEEDS THRESHOLD exception when an open online PRILOG record has exceeded the specified percentage of the maximum 16 MB record size.

This exception can have a Warning, Critical, or Dead status.

The PRILOG Record Size % parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST PRILOG solution to correct this exception.

RAA TOTAL FREE SPACE TOO SMALL exception

When the minimum total percentage of free space in the RAA falls below the percentage that is specified for the RAA Total Free Space % threshold, this exception entry is critical (red). This exception never has a warning (yellow) status or a dead (black) status.

The RAA Total Free Space % threshold is associated with this exception.

RECLAIMABLE FREESPACE EXCEEDED exception

The percentage of space occupied by deleted segments that can be reclaimed by reorganizing the database has exceeded the value specified for the Reclaimable Freespace Exceeded % threshold.

Note: This exception applies only to index databases. For HD databases, space occupied by deleted segments is returned to the database's free space and can be reused without reorganizing the database. However, for index databases, deleted segments can be retained and marked for deletion. The space is not returned to the database's free space for reuse until reorganization.

RECLAIMABLE IOVF EXCEEDED exception

When the specified percentage of the minimum IOVF that is reclaimable meets or exceeds the threshold, this exception entry is dead (black).

The solution to this exception is to review the ROOT and UOW parameters in the DBD.
RECLAIMABLE SPACE EXCEEDED exception

The number of bytes occupied by deleted segments that can be reclaimed by reorganizing the database has exceeded the value specified for the Reclaimable Bytes Exceeded threshold.

*Note*

This exception applies only to index databases. For HD databases, space occupied by deleted segments is returned to the database's free space and can be reused without reorganizing the database. However, for index databases, deleted segments can be retained and marked for deletion. The space is not returned to the database's free space for reuse until reorganization.

RECOVERY NEEDED COUNT INVALID exception

Recovery Advisor reports the RECOVERY NEEDED COUNT INVALID exception when the recovery needed count for an entity (DBD, DEDB, or HALDB partition) does not match the number of recovery needed flags turned on for the entity’s data set groups (DSGs) or areas.

The *Recovery needed count is invalid* parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBD solution to correct this exception.

TOO FEW DOVF CONTROL INTERVALS exception

When the percentage of DOVF CIs meets or exceeds the specified threshold percentage of the size of a UOW, this exception entry is dead (black).

The solution to this exception is to review the ROOT and UOW parameters in the DBD.

TOO FEW EMPTY RAPS exception

The percentage of empty RAPs in the RAA meets or falls below the value specified for the Too Few Empty RAPs % threshold.
**TOO FEW IOVF BLOCKS exception**

When the total number of IOVF blocks, specified as a percent of the size of the DEDB area (minus SDEP blocks), meets or exceeds the established threshold, this exception entry is dead (black).

The solution to this exception is to review the ROOT and UOW parameters in the DBD.

**TOO MANY CA SPLITS exception**

When the number of control area (CA) splits for a key-sequenced data set (KSDS) meets or exceeds the threshold, this exception entry is critical (red).

The splits are calculated as a percentage of total CAs in the data set. Only information from the DATA component of a KSDS is evaluated. Because of VSAM space reclamation, these values can exceed 100 percent if the index has not been rebuilt for a long time.

The CA Splits threshold is associated with this exception.

**TOO MANY CI SPLITS exception**

When the number of control interval (CI) splits for a KSDS meets or exceeds the threshold, this exception entry is critical (red).

The splits are calculated as a percentage of total CIs in the data set. Only information from the DATA component of a KSDS is evaluated. Because of VSAM space reclamation, these values can exceed 100 percent if the index has not been rebuilt for a long time.

The CI Splits threshold is associated with this exception.

**TOO MANY EXTENTS exception**

When the number of extents for a data set meets or exceeds the threshold, this exception entry is critical (shown in red).

If the number of extents reaches the IMS maximum for the data set access method, the exception is dead (black) because no more extents can be added to the data set.
The following thresholds are associated with this exception:

- OSAM Extents
- VSAM Extents

**TOO MANY FRAGMENTED DATABASE RECORDS exception**

This exception entry is created by phase two of the analysis. Phase two is scheduled when the I/O Probability Growth % threshold is met or exceeded to verify that the database will benefit from reorganization. Phase two reads the database sequentially and compares each record to the Improvement % threshold.

The Improvement % threshold is the difference between the number of blocks that a database record spans and the theoretical number of blocks that the same record will require after a reorganization. If a database record is distributed across 100 blocks but should fit in 30 blocks, the record can achieve a 70 percent improvement from reorganization.

The number of records that meet or exceed the threshold and the number of total records in the database are maintained. To create a percentage of fragmented records in the database, a percentage is calculated at the end of phase two by dividing the number of roots that met or exceeded the threshold by the total number of database roots. This percentage is compared to the Fragmentation % threshold.

If the fragmentation level meets or exceeds the Fragmentation % threshold, this exception entry is critical (red) and the exception entry that was added to the Exception List when the I/O Probability Growth % threshold was met or exceeded remains on the Exception List. If only one or neither of the Improvement % and Fragmentation % thresholds is met or exceeded, the exception entry is removed from the Exception List, and the I/O Probability Growth % threshold is increased dynamically to reflect that the database does not require reorganization.

**Example**

A database has a Probability of I/O base value of 20 percent and has grown over time to 30 percent, reaching the threshold of a 50 percent increase in I/O probability growth. Phase two of analysis is performed. The database has 2 million roots. Phase two determines that 1 million roots can be improved by the Improvement % threshold. The fragmentation value is 50 percent (1 million divided by 2 million equals 50 percent). If the Fragmentation % threshold is 50 or less, the I/O PROBABILITY TOO HIGH exception remains on the Exception List and a TOO MANY FRAGMENTED DATABASE RECORDS exception entry is added to the Exception List. If the Fragmentation % threshold is greater than 50, the I/O Probability Growth % threshold is increased dynamically and the I/O PROBABILITY TOO HIGH exception entry is removed from the Exception List.
Note
Phase two of analysis is not always performed. Phase two is not required for root-only databases and is not performed if the database has met or exceeded the Split Segments % threshold.

A database may have an increase in I/O probability that does not require reorganization. Phase two detects this condition and eliminates unnecessary reorganizations. An example of a database that has an increase in I/O probability but does not require reorganization is a database with one very large segment type that completely fills a block. I/O probability increases each time one of these segments is inserted. The database grows with these new segments (and others). I/O probability increases slightly because of the other segments but mostly because of the large segments. Reorganization would do little to improve overall performance. Phase two determines that most records would not be improved by reorganization and dynamically resets the I/O growth threshold accordingly.

The following thresholds are associated with this exception:

- I/O Probability Growth %
- Improvement %
- Fragmentation %

TOO MANY ROOTS IN OVERFLOW exception

When the percentage of roots in overflow meets or exceeds the value that is specified for the Roots in Overflow Exceed threshold, this exception entry is critical (red).

You can stop Database Advisor from reporting this exception by specifying zero for the Roots in Overflow Exceed threshold.

The Roots in Overflow Exceed threshold is associated with this exception.

TOO MANY SEGMENT SPLITS exception

When the percentage of split segments (Total Split Segment % on the Database State report) meets or exceeds the threshold, this exception entry is critical (red).

Example
A database with 2 million segments contains 300,000 split segments, and the current state value for Total Split Segment % is 15 percent (300,000 divided by 2 million equals 15 percent). If the Split Segment % threshold is 15 or less, the exception entry is critical.
The Split Segment % threshold is associated with this exception.

**TOO MANY SPLITS ACROSS BLOCKS exception**

When the percentage of split segments that point to different blocks meets or exceeds the Split Segments % Diff Block threshold, this exception entry is critical (red).

The Split Segments % Diff Block threshold is associated with this exception.

**TOO MANY VOLUMES exception**

When the number of volumes for a data set meets or exceeds the threshold, this exception entry is critical (red).

The following thresholds are associated with this exception:

- OSAM Volumes
- VSAM Volumes

**UNALLOCATED DATASET exception**

When data collection detects that a required database data set is not allocated, the severity (status) of this exception is the value specified by the Unallocated Data Set parameter.

**UNINITIALIZED DATASET exception**

When data collection detects an uninitialized data set, the severity (status) of this exception is the value specified by the Uninitialized Dataset parameter.

**Analyzing Database Advisor exceptions**

To research Database Advisor exceptions that are reported for a database, you can access menu commands directly from the Exception List.
You can view and analyze database information such as base values and historical information. You can also view and edit parameters such as site values, thresholds, and goals.

Accessing menu commands

To access menu commands directly from the Exception List, right-click a row in the Detail table in the Exception List window and choose any menu command.

Figure 30 on page 163 shows the menu that is available for a database data set object and the submenu that is available from the Reports command.

Figure 30: Accessing menu commands from the Exception List
Reviewing reports

Reports contain current and historical data that can help you with exception research and analysis.

Current state reports reflect the state of the database when data was collected most recently.

If you have a license for Database Advisor, the state reports include a data forecast that lists the values that Database Advisor used to determine whether a threshold has been violated or is expected to be violated.

History reports are useful for researching data trends. The charting capability in history reports is a good tool for viewing data about specific parameters over a specified time period.

For more information about using reports, see “Using reports” on page 429.

Reviewing analysis parameters and schedules

An exception might be reported for an object because the associated threshold value is not suitable for the object.

To address this situation, you can adjust the threshold value and can repeat the analysis process. If a solution is not acceptable for a database, you can adjust the goals for the database.

In addition to reviewing parameter values, for Database Advisor exceptions you might want to view and reset the base values for a database. If an exception has been reported to the Exception List because the data collection schedules are too frequent (or not frequent enough), you might want to change the data collection interval or the data collection schedules.

For more information about reviewing analysis parameters and schedules, see “Customizing parameters and schedules” on page 225.

Reviewing base values

If you have a license for Database Advisor, base values are recorded for a database when the database is reorganized and reloaded.

The base values represent the optimal state of a database and are used during the Database Advisor analysis. When you are analyzing an exception, you may want to
review the current or historical base values for the database that contains the exception.

There also may be situations in which you may want to reset base values. For example, if you have just recovered a database you may want to reset base values.

**Viewing current base values**

You can view current base values in the Navigation window.

1. To view current base values, in the Navigation window right-click the object for which you want to view base values and choose **Base Values**.

The Base Values window (Figure 31 on page 165) is displayed. Current base values are displayed on the Current tab.

**Figure 31: Current tab (Base Values window)**

<table>
<thead>
<tr>
<th>Current</th>
<th>History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used Space (MB)</td>
<td>0</td>
</tr>
<tr>
<td>Available Space (MB)</td>
<td>36</td>
</tr>
<tr>
<td>Available Free Space (MB)</td>
<td>36</td>
</tr>
<tr>
<td>Number of Allocated Blocks</td>
<td>126</td>
</tr>
<tr>
<td>Number of Used Blocks</td>
<td>63</td>
</tr>
<tr>
<td>Number of Free Space Blocks</td>
<td>0</td>
</tr>
<tr>
<td>Number of Used Extents</td>
<td>1</td>
</tr>
<tr>
<td>Number of Available Extents</td>
<td>249</td>
</tr>
<tr>
<td>Number of Used Volumes</td>
<td>1</td>
</tr>
<tr>
<td>Number of Available Volumes</td>
<td>3</td>
</tr>
<tr>
<td>Percent of CI Splits</td>
<td>0</td>
</tr>
<tr>
<td>Percent of CA Splits</td>
<td>0</td>
</tr>
<tr>
<td>High Used RBA</td>
<td>129024</td>
</tr>
<tr>
<td>Total Freespace %</td>
<td>100</td>
</tr>
</tbody>
</table>

**Viewing historical base values**

Use the following procedure to view historical base values in the Navigation window.
1. In the Navigation window, right-click the object for which you want to view historical base values and choose **Base Values**.

The Base Values window (Figure 32 on page 166) is displayed.

**Figure 32: History tab (Base Values window)**

![Base Values window](image)

2. On the History tab, click **List**.

A list of the utility executions (Figure 32 on page 166) that have been run on the selected object is displayed.

3. Select a utility execution, and click **View**.
The History Base Values dialog box (Figure 33 on page 167) is displayed.

**Figure 33: History Base Values dialog box**

```
<table>
<thead>
<tr>
<th>Used Space (MB)</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available Space (MB)</td>
<td>36</td>
</tr>
<tr>
<td>Available Free Space (MB)</td>
<td>36</td>
</tr>
<tr>
<td>Number of Allocated Blocks</td>
<td>126</td>
</tr>
<tr>
<td>Number of Used Blocks</td>
<td>63</td>
</tr>
<tr>
<td>Number of Free Space Blocks</td>
<td>0</td>
</tr>
<tr>
<td>Number of Used Extents</td>
<td>1</td>
</tr>
<tr>
<td>Number of Available Extents</td>
<td>249</td>
</tr>
<tr>
<td>Number of Used Volumes</td>
<td>1</td>
</tr>
<tr>
<td>Number of Available Volumes</td>
<td>3</td>
</tr>
<tr>
<td>Percent of Cl Splits</td>
<td>0</td>
</tr>
<tr>
<td>Percent of CA Splits</td>
<td>0</td>
</tr>
<tr>
<td>High Used RBA</td>
<td>129024</td>
</tr>
<tr>
<td>Total Freespace %</td>
<td>100</td>
</tr>
<tr>
<td>Alloc Freespace %</td>
<td>100</td>
</tr>
<tr>
<td>Execution Date</td>
<td>2007/10/02</td>
</tr>
<tr>
<td>Execution Time</td>
<td>17:48:13.000</td>
</tr>
</tbody>
</table>
```

**Resetting base values**

Database Advisor lets you reset base values by viewing the historical base values for past utility executions and selecting a set of base values from the list. Base values are reset to the values for the selected utility execution.

**WARNING**

The circumstances under which you should reset base values must be exceptional. For example, you may want to reset base values after recovering a database. For guidance about resetting base values, contact BMC Customer Support.

To reset base values after running IDCAMS repro, IMS application loads, and IBM recovery (DFSURDB0), include the //CPC$INCL DD DUMMY DD statement in your job step. The presence of the DD statement causes base values to be reset the next time that data is collected for the DBDs (the next time that the ENDOFDAY function is executed).
To reset base values in the console

1 In the Navigation window, right-click the object for which you want to reset base values and choose **Analysis Options => Base Values**.

The Base Values window (Figure 32 on page 166) is displayed.

2 On the History tab, click **List**.

A list of the utility executions (Figure 33 on page 167) that have been run on the selected object is displayed.

3 Select a utility execution, and click **View**.

The History Base Values dialog box (Figure 33 on page 167) is displayed.

4 To reset base values for the selected object, click **Reset**.

The base values are reset and are used during analysis.

For more information about base values, see “Analyze process” on page 142.

Using the Exception Forecast report

If you have a license for Database Advisor, you can use the Exception Forecast report to project future exceptions that are based on the rate at which a database is approaching a threshold.

The Exception Forecast report can be generated for specific date ranges and exception status types. You can export the data for use in third-party spreadsheets and management presentations.

Generating an Exception Forecast report

You can generate an Exception Forecast report in the Navigation window.

- IMS Database Administration folder
- Host
- IMSPLEX
- Database group
To generate an Exception Forecast report

1. In the Navigation window, right-click the IMS Databases object and select Reports => Exception Forecast.

The Exception Forecast Report Filter dialog box (Figure 34 on page 169) is displayed.

Figure 34: Exception Forecast Report Filter dialog box

Note
If you are generating an Exception Forecast report at any other level in the Navigation window, the report is available under the Reports menu.

2. In the Exception Forecast Report Filter dialog box, perform the following steps:
   a. Select a date range for the report.
   b. Select exception status types to include in the report.

3. Click OK.
The Exception Forecast window (Figure 35 on page 170) is displayed.

**Figure 35: Exception Forecast window**

![Exception Forecast window screenshot]

For more information about the Exception Forecast report, including how to export the report, see the online Help.

**Resetting Exception Forecast report criteria**

You can change report criteria, using the Exception Forecast Filter dialog box.

1. In the Exception Forecast window, click **Reset**.

   The Exception Forecast Report Filter dialog box (Figure 34 on page 169) is displayed.

2. Select a different date range or different exception status types to include in the report.

**Auditing databases to ensure analysis**

You can audit your databases to verify that data has been collected and analyzed as expected.

You can use the following methods to request an audit:
The stand-alone DLIAAUDR program creates a batch report that lists potential problems, but the program does not report exceptions to the Exception List.

The AUDIT function reports problems as exceptions in the Exception List, and also writes a report to the CPCLOG.

BMC recommends that you run the DLIAAUDR program first to identify existing problems, address those problems, and then begin running the AUDIT function on a regular basis.

Each audit verifies all databases that have a defined collection schedule as follows:

- The parameter that enables capturing statistics is set to Yes.
- The parameter that defines the collection interval is set to a nonzero value.

If the analysis is overdue by five or more days, the audit reports a potential problem.

### Reporting audit problems in a batch report

The DLIAAUDR program produces a batch report that is similar to the information that the AUDIT function reports to the CPCLOG data set.

#### To request an audit with DLIAAUDR

1. From member CPCAUDT in the sample library, obtain the sample JCL for executing DLIAAUDR.

2. In the JCL, change the CPCSYSIN options as needed to control the information that will appear in the report:

<table>
<thead>
<tr>
<th>CPCSYSIN option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOT_COLLECTED DAYS(n)</td>
<td>Report a problem for a database if the number of days from the scheduled collection date to the current date is greater than or equal to this specified number of days.</td>
</tr>
<tr>
<td>COLLECT_NOT_ACT</td>
<td>Report a problem for a database if the option to capture statistics is set to No or if the collection interval is set to 0.</td>
</tr>
<tr>
<td>ANALYZE_NOT_ACT</td>
<td>Report a problem for a database if the option to analyze the database is set to No.</td>
</tr>
<tr>
<td>NOT_ANALYZED DAYS(n)</td>
<td>Report a problem for a database if the number of days from the scheduled analysis date to the current date is greater than or equal to this specified number of days.</td>
</tr>
</tbody>
</table>

3. Submit the JCL.
Reporting audit problems as exceptions

You can request the AUDIT function through the Advisor scheduling service in the console or through the batch CPCTSCHD program.

The AUDIT function reports a potential problem as the highest-level Not Analyzed (orange) exception for the database in the Exception List.

AUDIT also produces a report in the CPCLOG data set of the Advisor server. The report indicates conditions that you might need to review:

■ Analysis flag off
■ Analysis overdue
■ Capture statistics flag off
■ Data collection overdue
■ Database collection not scheduled

To request the AUDIT function through the console

1 In the navigation window, right-click the IMSPLEX object and choose Configuration => Schedules.

2 In the Schedules window, double-click the AUDIT row in the table.

3 In the Edit Schedules dialog box, specify the schedule parameters.

4 Click Save.

To request the AUDIT function with program CPCTSCHD

1 Create a job to execute program CPCTSCHD, as explained in “Using program CPCTSCHD to initiate Advisor scheduled tasks” on page 289.
Using Recovery Advisor

This chapter contains information about working with Recovery Advisor exceptions and the Exception List.

Overview of Recovery Advisor processes

The Recovery Advisor collection and analysis process, known as Recovery Collect, detects exceptions that are based on the Recovery and RECON threshold parameters that you have established. When Recovery Collect is complete, Recovery Advisor reports exceptions in the exception list and recommends solutions to correct the reported problems.

This section explains how Recovery Advisor processes work and how you interact with those processes.

Note

This section applies only to Recovery Advisor. Database Advisor processes differ from Recovery Advisor processes in important ways. For more information, see Overview of Database Advisor processes on page 139.

Analyzing RECON records and statistics against threshold values

At the beginning of execution, Recovery Collect clears any previous Recovery Advisor exceptions (Database Advisor exceptions and other exceptions are not cleared).

Then the process reads records in the RECON and catalog statistics about the RECON data sets. The information is analyzed against established Recovery and RECON threshold values to find exceptions for registered databases, change accumulation data sets, log data sets, and RECON data sets. When an exception is found, Recovery Advisor records the exception in self-managing repositories, where any user of the console can access the information.
Unlike most Database Advisor exceptions, which can be used to track and predict trends, Recovery exceptions have no historical component. A Recovery exception exists or does not exist when Recovery Collect is performed. No history is kept for Recovery exceptions, and exception forecasting does not apply.

**Reporting exceptions in the Exception List window**

When Recovery Collect is complete, detected exceptions are available for display in the Exception List window.

The Exception List window displays all objects for which an exception was reported. When you select an object, the exceptions that Recovery Advisor has detected are displayed with exceptions that Database Advisor has detected for that object.

For each execution of Recovery Collect, a notification message is generated when E-mail/WTO parameters are enabled and when recovery or RECON exceptions are detected. Messages can be sent to registered recipients immediately or consolidated into one message that is sent on a scheduled basis.

**Recommending solutions, and generating JCL**

When you select an object in the Exception List window and click **Solutions**, the Solutions window is displayed.

This window lists the recommended solutions to correct detected exceptions, and you can request the display of solutions to correct selected exceptions. Solutions are associated with scripts that are used to customize skeleton JCL for executing functions or utilities (such as Image Copy or Recovery). When the customized JCL is generated and executed, the problems that are indicated by the exceptions are corrected.

**Using Recovery Advisor to manage Recovery thresholds**

To use Recovery Advisor to manage Recovery thresholds, perform the following general tasks.

1. Set parameter values in the Maintain Parameters window.
To configure Recovery Advisor for Recovery thresholds management, you establish parameter values in many parameter categories of the Maintain Parameters window. For more information, see “Establishing Recovery threshold parameters” on page 175.

2 Submit Recovery Collect for execution.

For more information, see “Executing Recovery Collect” on page 175.

3 Recognize that a Recovery threshold exception has been detected.

You can check the Exception List manually to look for reported Recovery threshold exceptions, or you can receive automated notification that a Recovery threshold exception has been detected. For more information, see “Working with Recovery Advisor exceptions” on page 178.

4 Use solutions to correct reported exceptions.

For more information, see “Working with Recovery Advisor solutions” on page 194.

Establishing Recovery threshold parameters

Recovery Advisor examines IMS RECON data sets for an IMSPLEX to find exception conditions, which indicate problems that can have an adverse affect on database operations, recoveries, and RECON availability and performance.

You can control the detection of Recovery threshold exceptions by establishing Recovery thresholds in the Maintain Parameters window. You can set recovery thresholds at host, IMSPLEX, and database group levels. For more information, see “Recovery Thresholds parameters” on page 242.

For general information about working with parameters, see “Customizing parameters” on page 226.

Executing Recovery Collect

You can execute Recovery Collect in any of the following ways:

- You can execute it automatically by setting up a RECOVERY COLLECT schedule. RECOVERY COLLECT controls the frequency and timing of the execution. Set the schedule to execute often enough to reflect changing conditions in your IMSPLEX. For more information, see “Customizing schedules” on page 279.
You can execute it on demand from the console by right-clicking an IMSPLEX object or a database group object and choosing Run Recovery Collect. Recovery Collect is performed only for the selected object.

For more information, see “Requesting Recovery Collect on demand” on page 176

You can use a batch job to execute Recovery Collect. With this method, you can specify an alternate RECON (a KSDS copy of your production RECON) for Recovery Advisor to use for detecting exceptions.

For more information, see “Using a batch job to execute Recovery Collect” on page 176

### Requesting Recovery Collect on demand

To initiate Recovery Collect on demand from the console, perform the following steps:

1. In the Navigation window, right-click an IMSPLEX or database group object.
2. From the menu, choose **Run Recovery Collect**.

   If this command is dimmed, your user ID does not have UPDATE access to the BBM.SDBA.IMSDB.*.SCD resource type.
3. In the Recovery Collect popup indicating that you are about to submit a request for a batch procedure, click **Yes**.

   Recovery Collect is executed for the IMSPLEX or database group that you have selected.

### Using a batch job to execute Recovery Collect

If you use a batch job to execute Recovery Collect (program IRMADVSE), Recovery Collect can process a copy of the "live" RECON data set (a copy might be required by your processing procedures).

**WARNING**

If you specify an alternate RECON, Recovery Advisor ignores RECON thresholds and does not discover RECON data set exceptions. Recovery Advisor detects normal recovery exceptions by using the status and information in the copy of the "live" RECON.
Coding JCL for program IRMADVSE

Member BRIADVSE of the product sample library contains a sample job that copies your RECON and executes program IRMADVSE.

Figure 36 on page 177 shows a job step that executes IRMADVSE.

Figure 36: IRMADVSE JCL

```plaintext
//jobname JOB your standard job statement
//BRIADVSE EXEC PGM=IRMADVSE,PARM='RECON,AUTORUN'
//STEPLIB DD DISP=SHR,DSN=your.BMC.load      * MUST BE APF AUTHORIZED
//         DD DISP=SHR,DSN=your.cpc.options.lib
//         DD DISP=SHR,DSN=your.RESLIB
//SYSPRINT DD SYSOUT=*                         
//SYSUDUMP DD SYSOUT=*                         
//RECON DD DISP=SHR,DSN=your.imsplex.recon     
//ALTRCN DD DISP=SHR,DSN=copyof.your.imsplex.recon
/*
```

Table 8 on page 177 describes statements in an IRMADVSE job step.

Table 8: IRMADVSE statements

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td>(Required) Specify PGM=IRMADVSE,PARM='RECON,AUTORUN'</td>
</tr>
<tr>
<td>STEPLIB DD</td>
<td>(Required) In the STEPLIB concatenation, include the following data sets:</td>
</tr>
<tr>
<td></td>
<td>■ Library that contains program IRMADVSE (your BMC product load library)</td>
</tr>
<tr>
<td></td>
<td>■ Library that contains your Cross Product Connectivity (CPC) options module (your CPC load library)</td>
</tr>
<tr>
<td></td>
<td>■ RESLIB that contains modules DFSCNV0 and DFSVC000</td>
</tr>
<tr>
<td></td>
<td>All libraries in this concatenation must be APF authorized.</td>
</tr>
<tr>
<td>RECON DD</td>
<td>(Required) Use this statement to identify the data set name of a RECON that is defined to Advisor in the IMSPLEX record. Recovery Advisor uses the name of this RECON to identify the threshold values to use and to store exceptions in the repositories.</td>
</tr>
<tr>
<td>ALTRCN DD</td>
<td>(Optional) You can use this statement to identify the data set name of an alternate RECON, which is a copy of the RECON that is identified by the RECON DD statement. If you specify this statement, Recovery Advisor processes this data set to detect exceptions so that the original RECON is not affected by the analysis process.</td>
</tr>
<tr>
<td></td>
<td><strong>Warning:</strong> If you specify an alternate RECON, Recovery Advisor does not discover RECON exceptions.</td>
</tr>
</tbody>
</table>
**Checking the return code**

The following table lists possible return codes for program IRMADVSE.

**Table 9: IRMADVSE return codes**

<table>
<thead>
<tr>
<th>Return code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>No exceptions were detected during this function execution.</td>
</tr>
<tr>
<td>04</td>
<td>One or more warning conditions were detected, but the program continued to completion.</td>
</tr>
<tr>
<td>08</td>
<td>One or more exceptions were detected.</td>
</tr>
<tr>
<td>14</td>
<td>The function detected an error condition that was related to the RECON data sets. For example, one error condition is that a RECON data set has been discarded.</td>
</tr>
<tr>
<td>16</td>
<td>An error with the PARM or a DD statement was detected, or an abend condition occurred.</td>
</tr>
</tbody>
</table>

**Working with Recovery Advisor exceptions**

Recovery Advisor can detect and report the recovery threshold exceptions that are described in this section.

For general information about handling exceptions, see “Handling exceptions” on page 295.

**BACKOUT NEEDED exception**

Recovery Advisor reports the BACKOUT NEEDED exception when an object is in BACKOUT-NEEDED status in the RECON.

This exception has a Dead status.

The **Backout is needed** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBDS solution to correct this exception.
CA DATA SET NOT CATALOGED exception

Recovery Advisor reports the CA DATA SET NOT CATALOGED exception when a change accumulation data set that is registered in the RECON is not cataloged in the system catalog.

The Additional Information field in the Details pop-up window lists the time period that was specified for the verification.

This exception has a Critical status.

The Assets entry is not cataloged parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST CAGRP solution to correct this exception.

CA RUN IN ERROR exception

Recovery Advisor reports the CA RUN IN ERROR exception when a change accumulation data set is marked as IN-ERROR in the RECON.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification.

This exception has a Critical status.

The Assets entry is not cataloged parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST CAGRP solution to correct this exception.

DBDS IN MULTIPLE DBDS GROUPS exception

Recovery Advisor reports the DBDS IN MULTIPLE DBDS GROUPS exception when an object is included in more than one database data set group.

The Additional Information field in the Details pop-up window lists the following information:

- Count of database data set groups in which Recovery Advisor found the object
- Names of a maximum of the first five groups

This exception has a Warning status.
The **Object is in multiple DBDS groups** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDGRP REMOVE solution to correct this exception.

**DBDS NOT IN ANY DBDS GROUP exception**

Recovery Advisor reports the DBDS NOT IN ANY DBDS GROUP exception when an object is not included in any database data set group.

This exception has a Warning status.

The **Object is not in any DBDS group** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDGRP ADD solution to correct this exception.

**DBDS NOT IN CHANGE ACCUM GROUP exception**

Recovery Advisor reports the DBDS NOT IN CHANGE ACCUM GROUP exception when an object is not included in any change accumulation group.

This exception has a Warning status.

The **Object is not in any CA group** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE CAGRP ADD solution to correct this exception.

**EEQE(S) FOUND exception**

Recovery Advisor reports the EEQE(S) FOUND exception when an I/O failure occurred on the media that is storing the object (the object is in EQE FOUND status in the RECON).

The **Additional Information** field in the Details pop-up window lists the number of EEQEs that Recovery Advisor found.

This exception has a Critical status.

The **EEQE(S) are detected** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECOVER TO CURRENT solution to correct this exception.
GENMAX BELOW THRESHOLD exception

Recovery Advisor reports the GENMAX BELOW THRESHOLD exception when the RECON GENMAX value for an object is less than the specified number.

The Additional Information field in the Details pop-up window lists the GENMAX value that Recovery Advisor found in the RECON and the GENMAX value that Recovery Advisor used for verification.

This exception has a Warning status.

The Verify GENMAX value parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS GENMAX solution to correct this exception.

GENMAX EXCEEDED THRESHOLD exception

Recovery Advisor reports the GENMAX EXCEEDED THRESHOLD exception when the RECON GENMAX value for the object is greater than the specified number.

The Additional Information field in the Details pop-up window lists the GENMAX value that Recovery Advisor found in the RECON and the GENMAX value that Recovery Advisor used for verification.

This exception has a Warning status.

The Verify GENMAX value parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS GENMAX solution to correct this exception.

GRPMAX BELOW THRESHOLD exception

Recovery Advisor reports the GRPMAX BELOW THRESHOLD exception when the RECON GRPMAX value for the change accumulation group is less than the specified number.

The Additional Information field in the Details pop-up window lists the GRPMAX value that Recovery Advisor used for verification.

This exception has a Warning status.
The **Verify GRPMAX value** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE CAGRP GRPMAX solution to correct this exception.

### GRPMAX EXCEEDED THRESHOLD exception

Recovery Advisor reports the GRPMAX BELOW THRESHOLD exception when the RECON GRPMAX value for the change accumulation group is greater than the specified number.

The **Additional Information** field in the Details pop-up window lists the GRPMAX value that Recovery Advisor used for verification.

This exception has a Warning status.

The **Verify GRPMAX value** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE CAGRP GRPMAX solution to correct this exception.

### HALDB PARTITION NOT INITIALIZED exception

Recovery Advisor reports the HALDB PARTITION NOT INITIALIZED exception when an object is a high availability large database (HALDB) that has not been initialized.

This exception has a Warning status.

The **HALDB partition is not initialized** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBDS solution to correct this exception.

### IC DATA SET IN ERROR exception

Recovery Advisor reports the IC DATA SET IN ERROR exception when an image copy data set is marked as IN-ERROR in the RECON.

The **Additional Information** field in the Details pop-up window lists the following information:

- Type (BATCH, ONLINE, CONCUR, SMSCIC, SMSNOC, or USER) and timestamp (in yyyy.ddd.hh:mm:ss.t format) of the image copy that is in error and that Recovery Advisor found in the RECON.
■ Time period that was specified for verification

This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the COPY IMAGE COPY solution to correct this exception.

### IC DATA SET NOT CATALOGED exception

Recovery Advisor reports the IC DATA SET NOT CATALOGED exception when an image copy data set that is registered in the RECON is not cataloged in the system catalog.

The **Additional Information** field in the Details pop-up window lists the following information:

■ Type (BATCH, ONLINE, CONCUR, SMSCIC, SMSNOC, or USER) and timestamp (in `yyyy.ddd.hh:mm:ss.t` format) of the image copy that is not cataloged and that Recovery Advisor found in the RECON

■ Time period that was specified for verification

This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the COPY IMAGE COPY solution to correct this exception.

### IC RECOMMENDED BY CHANGE ACCUM exception

Recovery Advisor reports the IC RECOMMENDED BY CHANGE ACCUM exception when the Change Accumulation utility determines that an object should be image copied.

This determination is based on IC Triggering by CA threshold values (as described in “Using the IC Triggering by CA feature” on page 385). The **Additional Information** field in the Details pop-up window lists the following information:

■ Size of the CA file, as determined during the CA execution

■ Number of objects in the CA group that were selected for an image copy

This exception has a Warning status.
The **IC Triggering by CA** threshold parameters control detection of this exception during execution of the BMC Change Accumulation utility. The Solution manager recommends the IC TRIGGERING IMAGE COPY solution, which specifies the SMARTCA(Y) keyword, to correct this exception.

**IMAGE COPY NEEDED exception**

Recovery Advisor reports the IMAGE COPY NEEDED exception when an object requires an image copy (it is in IC-NEEDED status in the RECON).

The **Additional Information** field in the Details pop-up window lists the following information:

- Type (BATCH, ONLINE, CONCUR, SMSCIC, SMSNOC, or USER) and timestamp (in yyyy.ddd.hh:mm:ss.t format) of the image copy that is not cataloged and that Recovery Advisor found in the RECON
- Time period that was specified for verification

This exception has a Critical status.

This exception can be reported if you do not have a license for the Backup and Recovery Solution *for IMS* product and you schedule Recovery Advisor to run.

The **Image copy is needed** parameter controls Recovery Advisor detection of this exception. If you have a license for the Backup and Recovery Solution *for IMS* product, the Solution manager recommends the BATCH/CONCURRENT IMAGE COPY solution to correct this exception.

**IMAGE COPY RECOMMENDED exception**

Recovery Advisor reports the IMAGE COPY RECOMMENDED exception when an object should be image copied (it is in IC-RECOMMENDED status in the RECON).

This exception has a Warning status.

The **Image copy is recommended** parameter controls Recovery Advisor detection of this exception. Depending on the Database Availability goal parameter, the Solution manager recommends the BATCH IMAGE COPY solution or the CONCURRENT IMAGE COPY solution to correct this exception.
MIN # OF CA RUNS NOT AVAILABLE exception

Recovery Advisor reports the MIN # OF CA RUNS NOT AVAILABLE exception when fewer than the specified number of valid change accumulation run records are registered for the change accumulation group.

The Additional Information field in the Details pop-up window lists the following information:

■ Value that Recovery Advisor used for verification (threshold parameter value or established GRPMAX value)

■ Time period that Recovery Advisor used for verification (if a time period was specified)

This exception has a Warning status.

The Number of CA runs is less than required parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE ACCUMULATION solution to correct this exception.

MIN # OF IC'S NOT AVAILABLE exception

Recovery Advisor reports the MIN # OF IC'S NOT AVAILABLE exception when fewer than the required number of valid image copies are registered for the object.

The Additional Information field in the Details pop-up window lists the following information:

■ Count of available image copies that Recovery Advisor found in the RECON

■ Value that Recovery Advisor used for verification (threshold parameter value or established GENMAX value)

■ Time period that Recovery Advisor used for verification (if a time period was specified)

This exception has a Warning status.

The Number of valid image copies is less than required parameter controls Recovery Advisor detection of this exception. Depending on the Database Availability goal parameter, the Solution manager recommends the BATCH IMAGE COPY solution or the CONCURRENT IMAGE COPY solution to correct this exception.
MORE LOGS THAN THRESHOLD exception

Recovery Advisor reports the MORE LOGS THAN THRESHOLD exception when more than the specified maximum number of logs has been written since the last image copy or change accumulation.

The Additional Information field in the Details pop-up window lists the following information:

- Count of logs that would be required to perform a recovery to current
- Type (image copy or change accumulation) and timestamp (in yyyy.ddd.hh:mm:ss.t format) of the most recent image copy or change accumulation data set that Recovery Advisor found in the RECON

This exception has a Warning status.

The Number of logs since last IC or CA exceeds a value parameter controls Recovery Advisor detection of this exception. Depending on the Database Availability goal parameter, the Solution manager recommends the BATCH IMAGE COPY solution or the CONCURRENT IMAGE COPY solution to correct this exception.

NO AREA DATA SET AVAILABLE exception

Recovery Advisor reports the NO AREA DATA SET AVAILABLE exception when no area data set (ADS) is available for a Fast Path area.

This exception has a Critical status.

The No area data set is available parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBDS solution to correct this exception.

NO AREA DATA SET DEFINED exception

Recovery Advisor reports the NO AREA DATA SET DEFINED exception when no ADS has been defined for a Fast Path area.

This exception has a Warning status.
The **No area data set is defined** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBDS solution to correct this exception.

### NO CA RUN WITHIN TIME RANGE exception

Recovery Advisor reports the NO CA RUN WITHIN TIME RANGE exception when CAGRP does not have a valid CA run within the specified time range.

The **Additional Information** field in the Details pop-up window indicates one of the following conditions:

- No CA run is available.
- One or more CA runs is in error. The field also indicates the time period that was specified.
- No CA runs were found within the time period that was specified.

This exception has a Warning status.

The **No CA run exists** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE ACCUMULATION solution to correct this exception.

### NO IC WITHIN TIME RANGE exception

Recovery Advisor reports the NO IC WITHIN TIME RANGE exception when a valid image copy of the object has not completed successfully within the specified number of hours or days.

The **Additional Information** field in the Details pop-up window lists the type (BATCH, ONLINE, CONCUR, SMSCIC, SMSNOC, or USER) and timestamp (in `yyyy:ddd.hh:mm:ss.t` format) of the most recent valid image copy that Recovery Advisor found in the RECON.

This exception has a Warning status.

This exception can be reported if you do not have a license for the Backup and Recovery Solution for IMS product and you schedule Recovery Advisor to run.

The **No valid image copy exists** parameter controls Recovery Advisor detection of this exception. Depending on the Database Availability goal parameter, the Solution
manager recommends the BATCH IMAGE COPY solution or the CONCURRENT IMAGE COPY solution to correct this exception.

**NO SECIC WITHIN TIME RANGE exception**

Recovery Advisor reports the NO SECIC WITHIN TIME RANGE exception when a primary image copy without a corresponding valid secondary image copy was found for an object within the specified number of hours or days.

The *Additional Information* field in the Details pop-up window lists the following information:

- Type (BATCH, ONLINE, CONCUR, SMSCIC, SMSNOC, or USER) and timestamp (in *yyyy.ddd.hh:mm:ss.t* format) of the most recent primary image copy for which Recovery Advisor did not find a valid secondary image copy
- Count of missing secondary image copies
- Count of primary image copies that Recovery Advisor found for the object

This exception has a Warning status.

The *No secondary image copy exists* parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the COPY IMAGE COPY solution to correct this exception.

**NO SECLOG WITHIN TIME RANGE exception**

Recovery Advisor reports the NO SECLOG WITHIN TIME RANGE exception when a PRILOG entry does not have a corresponding SECLOG entry within a specified time period.

The *Additional Information* field in the Details pop-up window lists whether an entry is missing or in error and the time period that was specified for verification.

This exception has a Critical status.

The *No SECLOG entry exists* parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.
NO SECSLD WITHIN TIME RANGE exception

Recovery Advisor reports the NO SECSLD WITHIN TIME RANGE exception when a PRISLD entry does not have a corresponding SECSLD entry within a specified time period.

The **Additional Information** field in the Details pop-up window lists whether an entry is missing or in error and the time period that was specified for verification.

This exception has a Critical status.

The **No SECSLD entry exists** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

NO VALID IMAGE COPY exception

Recovery Advisor reports the NO VALID IMAGE COPY exception when no valid image copy is registered in the RECON for the object.

The **Additional Information** field in the Details pop-up window indicates whether no image copy is available, whether one or more image copies are in error, or whether only user image copies are available.

This exception has a Critical status.

This exception can be reported if you do not have a license for the Backup and Recovery Solution for IMS product and you schedule Recovery Advisor to run.

The **No valid image copy exists** parameter controls Recovery Advisor detection of this exception. Depending on the Database Availability goal parameter, the Solution manager recommends the BATCH IMAGE COPY solution or the CONCURRENT IMAGE COPY solution to correct this exception.

PREOPEN NOT SET FOR AREA exception

Recovery Advisor reports the PREOPEN NOT SET FOR AREA exception when a Fast Path area does not have the PREOPEN option set in the RECON.

This exception has a Warning status.
The Area is not set to preopen parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS PREOPEN solution to correct this exception.

PRILOG ENTRY IN ERROR exception

Recovery Advisor reports the PRILOG ENTRY IN ERROR exception when a PRILOG entry in the RECON is marked as IN-ERROR.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification and whether batch logs were excluded from verification.

This exception has a Critical status.

The Assets entry is not cataloged parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

PRILOG ENTRY NOT CATALOGED exception

Recovery Advisor reports the PRILOG ENTRY NOT CATALOGED exception when a PRILOG entry in the RECON is not cataloged in the system catalog.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification and whether batch logs were excluded from verification.

This exception has a Critical status.

The Assets entry is not cataloged parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

PRISLD ENTRY IN ERROR exception

Recovery Advisor reports the PRISLD ENTRY IN ERROR exception when a PRISLD entry in the RECON is marked as IN-ERROR.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification.
This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

**PRISLD ENTRY NOT CATALOGED** exception

Recovery Advisor reports the PRISLD ENTRY NOT CATALOGED exception when a PRISLD entry in the RECON is not cataloged in the system catalog.

The **Additional Information** field in the Details pop-up window lists the time period that was specified for verification.

This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

**PROHIBIT AUTHORIZATION FOUND** exception

Recovery Advisor reports the PROHIBIT AUTHORIZATION FOUND exception when an object is in PROHIBIT- AUTHORIZATION status in the RECON.

This exception has a Critical status.

The **Object is in prohibit authorization status** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBDS solution to correct this exception.

**READ ONLY** exception

Recovery Advisor reports the READ ONLY exception when an object is in READ-ONLY status in the RECON.

This exception has a Warning status.

The **Object is in READ-ONLY status** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the START DATABASE solution to correct this exception.


RECOVERY NEEDED exception

Recovery Advisor reports the RECOVERY NEEDED exception when an object requires recovery (it is in RECOV-NEEDED status in the RECON).

This exception has a Dead status.

This exception can be reported if you do not have a license for the Backup and Recovery Solution for IMS product and you schedule Recovery Advisor to run.

The Recovery is needed parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECOVER TO CURRENT solution to correct this exception.

RECOVERY NEEDED COUNT INVALID exception

Recovery Advisor reports the RECOVERY NEEDED COUNT INVALID exception when the recovery needed count for an entity (DBD, DEDB, or HALDB partition) does not match the number of recovery needed flags turned on for the entity’s data set groups (DSGs) or areas.

The Recovery needed count is invalid parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST DBD solution to correct this exception.

RECOVPD BELOW THRESHOLD exception

Recovery Advisor reports the RECOVPD BELOW THRESHOLD exception when the RECOVPD value that is recorded in the RECON for an object is less than the specified number.

The Additional Information field in the Details pop-up window lists the RECOVPD value that Recovery Advisor found in the RECON and the RECOVPD value that Recovery Advisor used for verification.

This exception has a Warning status.

The Verify RECOVPD value parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS RECOVPD solution to correct this exception.
RECOVPD EXCEEDED THRESHOLD exception

Recovery Advisor reports the RECOVPD EXCEEDED THRESHOLD exception when the RECOVPD value that is recorded in the RECON for an object is greater than the specified number.

The Additional Information field in the Details pop-up window lists the RECOVPD value that Recovery Advisor found in the RECON and the RECOVPD value that RECOVERY ADVISOR used for the verification.

This exception has a Warning status.

The Verify RECOVPD value parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the CHANGE DBDS RECOVPD solution to correct this exception.

SECLOG ENTRY IN ERROR exception

Recovery Advisor reports the SECLOG ENTRY IN ERROR exception when a SECLOG entry in the RECON is in error.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification and whether batch logs were excluded from verification.

This exception has a Critical status.

The Assets entry is not cataloged parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

SECLOG ENTRY NOT CATALOGED exception

Recovery Advisor reports the SECLOG ENTRY NOT CATALOGED exception when a SECLOG entry in the RECON is not cataloged in the system catalog.

The Additional Information field in the Details pop-up window lists the time period that was specified for verification and whether batch logs were excluded from verification.

This exception has a Critical status.
The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

**SECSLD ENTRY IN ERROR exception**

Recovery Advisor reports the SECSLD ENTRY IN ERROR exception when a SECSLD entry in the RECON is in error.

The **Additional Information** field in the Details pop-up window lists the time period that was specified for verification.

This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

**SECSLD ENTRY NOT CATALOGED exception**

Recovery Advisor reports the SECSLD ENTRY NOT CATALOGED exception when a SECSLD entry in the RECON is not cataloged in the system catalog.

The **Additional Information** field in the Details pop-up window lists the time period that was specified for verification.

This exception has a Critical status.

The **Assets entry is not cataloged** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST LOG solution to correct this exception.

**Working with Recovery Advisor solutions**

To correct reported Recovery threshold exceptions, you can use Recovery Advisor solutions.

The solution process recommends the best solutions to correct problems that are reported by Recovery Collect. You can correct all (or certain) exceptions by using the recommended solutions or any available solutions.
The processes that Recovery Advisor uses to recommend solutions and generate JCL also apply to Database Advisor. If Recovery Advisor and Database Advisor report exceptions for the same object, the solution process chooses the best solutions to correct exceptions.

When you select a solution, the Generate JCL function generates JCL that is optimized for your environment. When the generated JCL is executed, the problems that are indicated by the exceptions are corrected.

Parameters that control solution processing include goal parameters, job control parameters, and parameters for specific functions.

For more information, see “Working with solutions, JCL generation, and viewing jobs” on page 323.

Clearing Recovery Advisor exceptions from the repository

Recovery Advisor exceptions are recorded in the repository during execution of Recovery Collect.

During initialization, Recovery Collect clears all existing Recovery Advisor exceptions from the repository so that any exceptions which are reported during the analysis process reflect current conditions in the RECON and current statistics for the RECON data sets.

When Recovery Collect is initiated by an on-demand request for a database group, Recovery Collect clears exceptions only for the objects in the group. Recovery Collect does not clear the IC RECOMMENDED BY CHANGE ACCUM exception. The BMC Change Accumulation utility creates the exception, and the BMC Image Copy utility clears it.

The RECON Reorg utility (which you can run as a solution) clears RECON exceptions when it completes successfully.

The BMC Image Copy utility, Recovery utility, and Change Accumulation utility also automatically clear certain Recovery Advisor exceptions from the repository, as explained in Table 10 on page 196. This action is accomplished only if the STEPLIB concatenation in the utility job step contains the Cross Product Connectivity (CPC) options module (member name CPCCOPT).
### Table 10: Recovery Advisor exceptions cleared by BMC utilities

<table>
<thead>
<tr>
<th>Exception</th>
<th>Utilities</th>
<th>Cleared when</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKOUT NEEDED</td>
<td>Recovery</td>
<td>Recovery completes successfully</td>
</tr>
<tr>
<td>EEQE(S) FOUND</td>
<td>Recovery</td>
<td>Recovery completes successfully</td>
</tr>
<tr>
<td>IC RECOMMENDED BY CHANGE ACCUM</td>
<td>Change Accumulation and Image Copy when SMARTCA(Y) is specified</td>
<td>At least one image copy is created successfully</td>
</tr>
<tr>
<td>IMAGE COPY NEEDED</td>
<td>Change Accumulation, Image Copy, Recovery</td>
<td>A least one image copy is created successfully</td>
</tr>
<tr>
<td>IMAGE COPY RECOMMENDED</td>
<td>Change Accumulation, Image Copy, Recovery</td>
<td>At least one image copy is created successfully</td>
</tr>
<tr>
<td>NO CA RUN WITHIN TIME RANGE</td>
<td>Change Accumulation</td>
<td>Change accumulation completes successfully</td>
</tr>
<tr>
<td>NO IC WITHIN TIME RANGE</td>
<td>Change Accumulation, Image Copy, Recovery</td>
<td>At least one image copy is created successfully</td>
</tr>
<tr>
<td>NO SECIC WITHIN TIME RANGE</td>
<td>Change Accumulation, Image Copy, Recovery</td>
<td>At least two output image copies are created successfully. This exception is also cleared when the Copy Image Copy function completes successfully.</td>
</tr>
<tr>
<td>NO VALID IMAGE COPY</td>
<td>Change Accumulation, Image Copy, Recovery</td>
<td>At least one image copy is created successfully</td>
</tr>
<tr>
<td>RECOVERY NEEDED</td>
<td>Recovery</td>
<td>Recovery completes successfully</td>
</tr>
</tbody>
</table>
Using RECON Advisor functions

This section contains information about using RECON Advisor functions.

RECON Advisor functions include:

- RECON thresholds, exceptions, and solutions
- RECON Reorg utility
- DBRC List History function

RECON management components

RECON thresholds, exceptions, and solutions work together with the RECON Reorg utility to help you manage your DBRC RECON data sets as follows:

- It determines when routine tasks should be performed.
- It detects problems that require immediate attention.
- It notifies you about RECON problems.
- It generates JCL to address detected conditions.

By performing routine tasks exactly when they are needed, rather than on a predetermined schedule, you can maintain the RECON at optimum availability and performance levels with best use of time and processing resources. By receiving timely notification of RECON problems, you can address them before they negatively affect your service level agreements. By using generated JCL, you save time and avoid errors.

During execution of Recovery Advisor, RECON threshold values that you have established are compared with current values to detect RECON exceptions. When you select a RECON for which an exception was detected, Advisor recommends the RECON UTILITIES solution to correct that exception. This solution generates JCL for the RECON Reorg utility. When you submit the generated JCL, the RECON Reorg utility corrects the problem.
Note

The components that support RECON management function in Advisor are available if you have a license for the Backup and Recovery Solution product for IMS or the MAXM Database Advisor product for IMS. You execute Recovery Advisor to obtain the support.

Recovery Advisor analyzes RECON exceptions as follows:

1. The first time that Recovery Collect is executed, Recovery Advisor establishes base values for the statistical information that is collected for the RECON data sets.

2. During subsequent executions of Recovery Collect, Recovery Advisor collects current statistics for the RECON data sets.

3. Recovery Advisor compares threshold values to current values and base values. Recovery Advisor determines whether a threshold value has been met or exceeded or, if not, the rate at which a current value is approaching the threshold value.

   - If a value is predicted to meet or exceed the threshold value within the number of warning days that you have specified, Recovery Advisor reports a warning exception.
   
   - If a value has met or exceeded a threshold, Recovery Advisor reports a critical exception.
   
   - If a value has reached the system upper limit, Recovery Advisor reports a dead exception.

Recovery Advisor also reports current statistics in the RECON State report. For more information, see “Evaluating the state of RECONs” on page 434.

DBRC List History function

The DBRC List History function is a GUI representation of the data the DBRC returns from the LIST.HISTORY command. You can view the data in a table or a timeline format.

In the GUI, you can click certain events and tabs to submit DBRC commands or generate JCL. For example, you can submit a command to change the GENMAX value for a database data set, or generate JCL to create an image copy or recover a database to a point in time.
Using Advisor for RECON management

You can configure Advisor for RECON management by establishing parameter values in the RECON Thresholds category, the RECON Thresholds/Operational Controls category, and the Job Control category of the Maintain Parameters window.

1 Set parameter values in the Maintain Parameters window.

To configure Advisor for RECON management, you establish parameter values in the RECON Thresholds category, the RECON Thresholds/Operational Controls category, and the Job Control category of the Maintain Parameters window. For more information, see “Establishing parameters for RECON management” on page 200.

2 Submit Recovery Advisor for execution by using one of the following methods:

- Schedule the RECOVERY COLLECT function to execute at predetermined times as described in “Customizing schedules” on page 279.
- Request execution of Recovery Collect on demand as described in “Requesting Recovery Collect on demand” on page 176.
- Execute Recovery Collect by using a batch job as described in “Using a batch job to execute Recovery Collect” on page 176.

3 Recognize that a RECON exception has been detected.

You can check the Exception List manually to look for reported RECON exceptions, or you can receive automated notification that a RECON exception has been detected. For more information, see “Recognizing RECON threshold exceptions” on page 203.

4 View recommended solutions.

From the Exception List, select a RECON object for which one or more exceptions were reported and click Solutions. The RECON UTILITIES solution is the recommended solution for RECON exceptions. For more information, see “Using the RECON UTILITIES solution” on page 206.

5 Use the RECON UTILITIES solution to generate JCL.

From the Solutions window, click Generate JCL. The generated JCL executes the RECON Reorg utility (program BRIURCN0). For more information, see “Using the RECON Reorg utility” on page 208.

6 Submit the generated JCL to execute the RECON Reorg utility, and check the results.
Establishing parameters for RECON management

To configure Advisor for RECON management, you establish parameter values in the RECON Thresholds category, the RECON Thresholds/Operational Controls category, and the Job Control category of the Maintain Parameters window.

Recovery Advisor compares the established RECON threshold values with current values to detect RECON exceptions. The RECON Reorg utility uses operational control and job control values to resolve contention problems and perform optional backups.

This section explains how these parameters are used. For general information about maintaining parameters, see “Customizing parameters” on page 226.

Parameters for setting RECON thresholds

Recovery Advisor always detects an unavailable or discarded RECON.

You can specify values for the following parameters to control whether and how Recovery Advisor detects other exceptions for RECON data sets:

- CI Splits %
- CA Splits %
- Alloc Freespace %
- Extents
- Volumes
- Data set size (tracks)
- PRILOG Record Size %

You set values for these parameters in the RECON Thresholds category of the Maintain Parameters window. You can set values at host and IMSPLEX levels. For more information, see “RECON Thresholds parameters” on page 247.

From the Solutions window, click Submit. For more information, see “Return codes of the RECON Reorg utility” on page 210.
Parameters for handling RECON contention

You can specify the following parameters to control how the RECON Reorg utility handles RECON contention during a reorganization of the RECON data sets:

- Resolution Option
- Retry Time Period
- Return Code

You set values for these options in the RECON Thresholds/Operational Controls category of the Maintain Parameters window. For more information, see “RECON Thresholds parameters” on page 247.

Resolution Option parameter

The Resolution Option parameter specifies the action that the RECON Reorg utility takes when it encounters RECON contention.

You can choose one of the following actions:

- Fail RECON utility with Return Code
  The utility job step fails with the return code that is specified with the Return Code parameter. This action is the default.

- Depend on RECON Loss Notification
  If you are using the Common Service Layer and the Automatic RECON Loss Notification service is available, the utility can use this DBRC service.

- WTOR to notify operators
  If you want your operators to determine which jobs are causing RECON contention and take care of them, choose this action. You can also specify the text of the WTOR message to be issued. For more information, see “Responding to a WTOR from the RECON Reorg utility” on page 212.

- RECOVERY MANAGER for IMS to notify IMS subsystems
  If you have a password for the Backup and Recovery Solution product or the RECOVERY MANAGER product for IMS, this option is available. The utility calls RECOVERY MANAGER to perform a /DIS OLDS command on each IMS subsystem. This command causes the IMS system to release its allocation of the discarded RECON.

- DELTA PLUS to notify IMS subsystems
  If you have a password for the DELTA PLUS product, this option is available. The utility calls DELTA PLUS to perform a /DIS OLDS command on each IMS subsystem. This command causes the IMS system to release its allocation of the discarded RECON.
- Fail RECON utility with Abend Code

  If you want the utility to fail with an abend code, choose this option and specify the abend code value that you want the utility to use.

**Retry Time Period parameter**

Regardless of the value of the Resolution Option parameter, the RECON Reorg utility continues to try to perform the reorganization for the duration that you have specified with the Retry Time Period parameter (the default is 20 minutes).

If the RECON contention still exists after the specified time period, the utility ends un成功fully.

**Return Code parameter**

If RECON contention is not resolved within the specified time period, the RECON Reorg utility fails with the return code that is specified as the value of the Return Code parameter. The default is 12.

**Parameter to request backups of RECON data sets**

The RECON Reorg utility can request backups of RECON data sets before performing actions that affect the data in the RECON.

You can use the RECON Backup Data Set Name parameter (in the RECON Thresholds/Operational Controls category of the Maintain Parameters window) to control whether the utility requests backups. For more information, see “RECON Thresholds parameters” on page 247.

If you specify a value for the Backup Data Set Name parameter, the utility calls DBRC with a BACKUP.RECON command. Any RECON data set that is discarded or unavailable is redefined before the backup is executed. If no RECON data set is discarded or unavailable, the backup is executed before any attempt to reorganize all RECON data sets (when the utility is in Reorg All mode). If the backup fails, the utility execution ends un成功fully.

The RECON UTILITIES solution generates JCL for the RECON Reorg utility. This solution uses the value of the Backup Data Set Name parameter as the value of the DSN operand of the BACKUP1 DD statement in the generated JCL. The specified data set must exist and must be in a format acceptable to DBRC. A DBRC backup data set can be a KSDS or a sequential data set. A sequential data set might be more convenient because it can be replaced and does not have to be empty.
Recognizing RECON threshold exceptions

When Recovery Advisor is executed, it compares established RECON threshold values with current values to detect and report RECON threshold exceptions.

This section explains RECON threshold exceptions. For general information about exceptions, see “Handling exceptions” on page 295.

DSN SIZE EXCEEDS THRESHOLD exception

Recovery Advisor reports the DSN SIZE EXCEEDS THRESHOLD exception when the established threshold number of tracks has been violated for a RECON data set.

The Additional Information field in the Details pop-up window cites the current number of tracks for the data set.

This exception has a Critical status when the threshold has been met or exceeded.

The Data set size (tracks) parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.

INSUFFICIENT FREE SPACE exception

Recovery Advisor reports the INSUFFICIENT FREE SPACE exception when the established threshold amount of free space (relative to the total amount of space) has been violated for a RECON data set.
The **Additional Information** field in the Details pop-up window cites the current percentage of free space for the data set.

This exception has a Critical status when the threshold has been met or has decreased below the established threshold.

The **Alloc Freespace %** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.

---

**PRILOG SIZE EXCEEDS THRESHOLD exception**

Recovery Advisor reports the PRILOG SIZE EXCEEDS THRESHOLD exception when an open online PRILOG record has exceeded the specified percentage of the maximum 16 MB record size.

This exception can have a Warning, Critical, or Dead status.

The **PRILOG Record Size %** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the LIST PRILOG solution to correct this exception.

---

**RECON DISCARDED / UNAVAILABLE exception**

Recovery Advisor reports the RECON DISCARDED / UNAVAILABLE exception when a RECON data set has a status of DISCARDED or UNAVAILABLE.

The **Additional Information** field in the Details pop-up window cites the RECON data set status.

This exception has a Dead status.

No parameter is used to control Recovery Advisor detection of this exception. (This exception is always detected.) The Solution manager recommends the RECON UTILITIES solution to correct this exception.

---

**TOO MANY CA SPLITS exception**

Recovery Advisor reports the TOO MANY CA SPLITS exception when the established threshold for control area (CA) splits has been violated for a RECON data set.
The **Additional Information** field in the Details pop-up window cites the percentage of CA splits, relative to the total number of CAs in the data set.

This exception has a Critical status when the threshold has been met or exceeded.

The **CA Splits %** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.

**TOO MANY CI SPLITS exception**

Recovery Advisor reports the TOO MANY CI SPLITS exception when the established threshold for control interval (CI) splits has been violated for a RECON data set.

The **Additional Information** field in the Details pop-up window cites the percentage of CI splits, relative to the total number of CIs in the data set.

This exception has a Critical status when the threshold has been met or exceeded.

The **CI Splits %** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.

**TOO MANY EXTENTS exception**

Recovery Advisor reports the TOO MANY EXTENTS exception when the established threshold for number of extents has been violated for a RECON data set.

The **Additional Information** field in the Details pop-up window cites the current number of extents for the data set.

This exception has a Critical status when the threshold has been met or exceeded.

The **Extents** parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.


TOO MANY VOLUMES exception

Recovery Advisor reports the TOO MANY VOLUMES exception when the established threshold for number of volumes has been violated for a RECON data set.

The Additional Information field in the Details pop-up window cites the current number of volumes for the data set.

This exception has a Critical status when the threshold has been met or exceeded.

The Volumes parameter controls Recovery Advisor detection of this exception. The Solution manager recommends the RECON UTILITIES solution to correct this exception.

Using the RECON UTILITIES solution

If Recovery Advisor reports an exception for a RECON object, the Solutions window lists the RECON UTILITIES solution as the recommended solution for that object.

This solution creates JCL to execute the RECON Reorg utility (program BRIURCN0). Depending on the type of exception, the generated JCL requests execution of the RECON Reorg utility in Replace mode or Reorg All mode.

You can also use the Generate JCL function to select RECON utilities and create JCL to reorganize all RECON data sets.

This section describes the RECON UTILITIES solution. For general information about solutions, see “Working with solutions, JCL generation, and viewing jobs” on page 323.

Replace mode JCL

If Recovery Advisor reports the RECON DISCARDED/UNAVAILABLE exception, the REPLACE RECON DATA SET feature of the RECON UTILITIES solution generates JCL for the RECON Reorg utility to execute in Replace mode.

Figure 37 on page 206 shows generated JCL for executing the utility in Replace mode.

Figure 37: Generated RECON Reorg utility JCL (Replace mode)

```bash
//jobname JOB <standard job statement parameters>

//******************************************************************
//*  GENERATED JCL CREATED ON 2008-01-24 AT 13:45
```
Reorg All mode JCL

If Recovery Advisor does not report the RECON DISCARDED/UNAVAILABLE exception, the REORG ALL RECON DATA SETS feature of the RECON UTILITIES solution generates JCL for the RECON Reorg utility to execute in Reorg All mode.

Figure 38 on page 207 shows generated JCL for executing the utility in Reorg All mode.

Figure 38: Generated RECON Reorg utility JCL (Reorg All mode)
Using the RECON Reorg utility

The RECON UTILITIES solution generates JCL to execute the RECON Reorg. This utility restores RECON data sets to optimum availability and performance levels and resets base values for the statistical information that is collected for the RECON data sets.

Modes of the RECON Reorg utility

The RECON Reorg utility can execute in Replace mode or Reorg All mode:

- In Replace mode, the utility discovers any RECON data set that is discarded or unavailable and calls IDCAMS to perform a delete and define of the data set. The utility calls DBRC with a LIST.RECON STATUS command to force a copy of the active RECON into the newly defined data set or to set the data set to SPARE.
status. If Replace mode is recommended, the RECON data set has a serious problem and you might want to run the utility as soon as possible.

■ In Reorg All mode, the utility systematically handles reorganizing each data set until the RECONs are ordered with RECON1 as COPY1, RECON2 as COPY2, and RECON3 as SPARE. To accomplish this action, the utility performs the DBRC command CHANGE.RECON REPLACE, followed by a call to IDCAMS to delete and define the newly discarded RECON data set.

In Reorg All mode, all RECON data sets are involved; therefore, it is important to run the utility when no batch processes are executing and online IMS activity is low. The utility must be able to obtain an exclusive enqueue on each RECON data set to be able to delete and redefine it. Several parameters in the RECON Thresholds/Operational Controls category control how the utility handles RECON contention conditions.

The RECON UTILITIES solution generates JCL for the utility to execute in the correct mode, depending on the type of exception that was reported for the RECON object.

Requirements for executing the RECON Reorg utility

The RECON Reorg utility has the following requirements:

■ Parameter values must be established as explained in “Parameters for handling RECON contention” on page 201 and “Parameter to identify IDCAMS DEFINE control statements” on page 203. You can also request RECON backups as explained “Parameter to request backups of RECON data sets” on page 202.

■ The user ID that is assigned to the submitted job (either by the system or by the JCL) must have Alter authority for the RECON data sets because they are deleted and defined within the utility job.

■ RECON dynamic allocation (MDA) members must be available. The RECON Reorg utility verifies that the RECON data set name in the associated RECON MDA member matches the RECON name in the exception. When generating JCL, the RECON UTILITIES solution creates an IMSDALIB DD statement in the utility JCL. This statement contains all MDALIB libraries and the RESLIB library that were defined to the IMSPLEX with which the RECON is associated. The utility searches for RECON MDA members in these libraries.

RECON Reorg utility JCL

The RECON UTILITIES solution generates JCL for executing the RECON Reorg utility.
The following statements are valid in this JCL:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td>On the EXEC statement, the PGM parameter defines the program to be executed (BRIURCN0). The PARM parameter defines the function to be executed (REPL for Replace mode or ALL for Reorg All mode) and the name of the RECON data set to be processed.</td>
</tr>
<tr>
<td>STEPLIB</td>
<td>The STEPLIB DD statement identifies the libraries that contain the program and the processing option values to be used during execution.</td>
</tr>
<tr>
<td>IMSDALIB DD</td>
<td>The IMSDALIB DD statement identifies the data sets that contain dynamic allocation (MDA) members for allocating the RECON data sets.</td>
</tr>
<tr>
<td>IDCAMINn DD</td>
<td>The IDCAMINn DD statements identify the data set and members that contain the IDCAMS DEFINE statements for defining the RECON data sets.</td>
</tr>
<tr>
<td>BACKUP1 DD</td>
<td>The BACKUP1 DD statement identifies the name of the data set that the RECON Reorg utility uses to request a backup of the RECON data sets. This statement is present only if you have specified a value for the Backup Data Set Name parameter in the Maintain Parameters window.</td>
</tr>
<tr>
<td>SYSIN DD</td>
<td>The SYSIN DD statement identifies a temporary file that is used by the utility.</td>
</tr>
<tr>
<td>SYSOUT DD</td>
<td>The SYSOUT DD statement contains a running log of utility actions.</td>
</tr>
<tr>
<td>DBRCOUT DD</td>
<td>The DBRCOUT DD statement contains the DBRC output that resulted from the commands which were issued to DBRC by the utility.</td>
</tr>
<tr>
<td>IDCPRINT DD</td>
<td>The IDCPRINT DD statement contains the IDCAMS output that resulted from the DELETE and DEFINE commands which were issued to IDCAMS by the utility.</td>
</tr>
</tbody>
</table>

### Clearing RECON exceptions and resetting statistics

When the RECON Reorg utility completes successfully, all RECON exceptions are cleared from the repository. When the utility does not complete successfully, no RECON exceptions are cleared.

When the utility is executing in Replace mode and completes successfully, RECON statistics are not reset in the repository. When the utility is running in Reorg All mode and completes successfully, RECON statistics are reset.

### Return codes of the RECON Reorg utility

The RECON Reorg utility can issue the following return codes:
### Table 6: Return Codes and Descriptions

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>The utility successfully replaced a RECON data set or was able to reorganize all RECON data sets. All RECON exceptions are cleared.</td>
</tr>
<tr>
<td>04</td>
<td>The utility encountered a warning condition. For example, the space for a RECON definition is too small and extents were used. The utility completes successfully, and all RECON exceptions are cleared.</td>
</tr>
<tr>
<td>08</td>
<td>The utility encountered an error condition while preparing to process the RECON data sets. See the SYSOUT data set for the accompanying error message. RECON exceptions are not cleared.</td>
</tr>
<tr>
<td>12</td>
<td>The utility encountered an error condition during processing of the RECON data sets. See the SYSOUT data set and the DBRCOUT data set for more information. RECON exceptions are not cleared.</td>
</tr>
</tbody>
</table>

**nn**

**nn** is the return code value that is specified by the Return Code parameter in the RECON Thresholds/Operation Controls category of the Maintain Parameters window. The utility was unable to resolve RECON contention. RECON exceptions are not cleared.

### SYSOUT information from the RECON Reorg utility

The SYSOUT data set contains information about the actions that were performed during execution of the RECON Reorg utility.

#### Figure 39: SYSOUT information from the RECON Reorg utility

```
RECON DATA SET STATUS - JOB: BRSA#H04                                 RUN: 01/17/2008 17:18:17.03
RECON1: COPY1             BMC.RECON1
RECON2: DISCARDED          BMC.RECON2
RECON3: COPY2             BMC.RECON3

INVOKING IDCAMS -         BMC.RECON2                                 RUN: 01/17/2008 17:18:19.44
IDCAMIN2 - ALTERNATE SYSIN DD
IDCAMOUT - ALTERNATE SYSPRINT DD
//IDCAMIN2 DD *
DELETE BMC.RECON2          PURGE
DEFINE CLUSTER -          
  (NAME(BMC.RECON2) -
   INDEXED -
   KEYS (32,0) -
   Recordsize (512 2000) -
   NOREUSE -
   UNORDERED -
   FREESPACE (40,40) -
   CISZ (2048) -
   FOR(9999) -
   SHAREOPTIONS(3 3) -
   RECOVERY -
   NOERASE) -
   DATA  (NAME(BMC.RECON2.DATA) -
   CYL(10 0) -
   VOL(DEV9XX)) -
   INDEX  (NAME(BMC.RECON2.INDEX) -
   TRK(1 1) -
   VOL(DEV9XX))

INVOKING DBRC -            RUN: 01/17/2008 17:18:21.44
DBRCIN - ALTERNATE SYSIN DD
DBRCOUT - ALTERNATE SYSPRINT DD
//DBRCIN DD *
LIST.RECON STATUS
BACKUP.RECON RECON1
```
Responding to a WTOR from the RECON Reorg utility

You can determine which jobs are causing RECON data set contention and take care of them. If you want your operators to determine which jobs are causing RECON data set contention and take care of them, specify the WTOR to notify operators value for the Resolution option parameter. You can also specify the text of the write-to-operator-with-reply (WTOR) message to issue.

A RECON contention problem occurs when the RECON Reorg utility needs to delete and define a RECON data set but detected that some other subsystem, job, or user has allocated the data set. The utility can send one or more WTOR messages to the system console:

- If the RECON Reorg utility is replacing a single RECON data set, the utility issues the WTOR message for that data set.
- If the utility is reorganizing all RECON data sets and if IMS subsystems are running, the utility issues the WTOR message for each individual data set in the set of RECONs. If other jobs or users have allocated the RECON data sets, the utility issues the WTOR for each RECON data set until any allocations are freed.

The utility waits for a response from an operator. The operator should enter a /DIS OLDS command on all appropriate IMS subsystems to free each subsystem’s allocation of the discarded RECON data set. The operator should also ensure that no other batch jobs (or TSO users) have allocated the RECON data set.

After the operator has resolved the data set contention, or if the operator needs to take further action before the utility can continue, the operator can reply with one of the following codes:

- C — Cancel the utility. The RECON Reorg utility terminates and leaves a RECON data set in DISCARDED status. Rerun the RECON Reorg utility after all RECON
contention from batch jobs has been eliminated. The utility then redefines the discarded data set and restores its status to SPARE.

- **R** — Retry for another period of time, which is specified by the Retry Time Period option.

- **S** — Show the RECON enqueues. This reply causes the utility to display a list of the enqueues and then redisplay the WTOR message.

**WARNING**

If the jobs that are causing RECON contention are batch IMS jobs, the operator must leave the WTOR outstanding until any JCL allocations of the RECON data sets have been freed. If the jobs that are causing RECON contention are IMS online regions, the operator must enter a /DIS OLDS command on each IMS subsystem to release each RECON data set allocation. When IMS processes the /DIS OLDS command, it releases the allocation of the individual RECON data set that the RECON Reorg utility is preparing to delete and redefine. The operator must respond to multiple WTORs if multiple RECON data sets are being reorganized.

---

**Generating JCL to prepare RECONs for disaster recovery**

You can use the Disaster Recovery RECON Cleanup (DRRCN) utility to prepare a DBRC RECON data set for IMS startup and database recovery. For complete information about the DRRCN utility, see the *Backup and Recovery Products for IMS Recovery Manager User Guide*.

**To generate JCL to prepare RECONs for disaster recovery**

1. From the Database Management console, right-click a RECON object in the navigation pane, and select **Generate JCL**.

2. From the Solution/Feature list, select the row that contains RECON UTILITIES and RECON CLEANUP FOR DISASTER RECOVERY.

3. Click **Generate JCL**.

   The console creates DRRCN JCL in CHECK mode. (The FUNCTION keyword value is set to CHECK.) In this mode, the utility produces a report but does not update the RECON copy.

4. After determining that the utility will produce the results you want, change the FUNCTION keyword value to UPDATE, and rerun the job.
Using the DBRC List History function

The DBRC List History function is a GUI representation of the data the DBRC returns from the LIST.HISTORY command. You can view the data in a table or a timeline format.

In the GUI, you can click certain events and tabs to submit DBRC commands or generate JCL. For example, you can submit a command to change the GENMAX value for a database data set, or generate JCL to create an image copy or recover a database to a point in time.

For more information, view the Quick Course "Recovery Advisor for IMS - Using List History."

Accessing the DBRC List History window

You use the DBRC List History function through the DBRC List History window in the Database Management Console.

Before you begin

To access the DBRC List History window, your minimum level of access to the DBRC RECONs must be READ. For the window to display information about the selected object, the object must be registered with DBRC.

To access the DBRC List History window

1. In the Navigation window, right-click an eligible object and choose DBRC List History.

   You can open the List History window for the following objects:
   - IMS full-function databases
   - HALDB partitions
   - DEDB areas

2. In the DBRC List History - Limit Time Range for Events dialog, select the option button that corresponds to one of the following options:
   - Show all events that occurred from the specified number of hours in the past until now.
Show all events that are recorded in this RECON. (For large RECONs, the interface might take longer to display the results because it must process more data.)

Show all events that occurred from the specified date and time in the past until now.

3 Click OK.

The interface sends a request to the server for List History data, and then formats and displays the returned data in the DBRC List History window. See “DBRC List History window” on page 215.

**DBRC List History window**

The DBRC List History window graphically represents the information that DBRC provides when you enter the LIST.HISTORY command.

*Figure 40: DBRC List History window (with timeline view)*
**DSG or area tabs**

If the object you select has multiple data set groups (DSGs), the interface displays a separate tab for each DSG.

Each tab (representing the information about a DSG or area) contains the following elements:

- **On the left side of the tab**, the event pane displays the occurrence of events graphically in either of the following formats:
  - **Timeline**
    Events are shown as icons within a horizontal representation of the flow of time. Older events are at the left of the timeline, with newer events at the right.
  - **Table**
    Each event is a row in a table. Older events are at the top of the table, with newer events at the bottom. This layout is similar to the output of the DBRC LIST.HISTORY command.

For more information, see “Events in the DBRC List History Window” on page 217.

- **On the right side**, the Details pane lists detailed information about a selected event and provides access to event-specific actions:
  - **The Event tab** contains the information that DBRC has recorded for the event.
  - **The DSG or Area tab** contains information about the data set group (DSG) or Fast Path area that is related to this event.
  - **The CA tab** identifies the change accumulation group to which the related DSG or area belongs.
  - **The Prilog tab** identifies the primary log that is related to this event.

**Actions**

You can work with elements in this window as follows:

- **Select an event** by clicking it in the event pane.

- **Along the bottom of the window**, check boxes control the types of events to display in the event pane. Clear a box to hide the corresponding type of event.

- **Resize, maximize, and hide** the event and Details panes by dragging or clicking the pane controls (located within the border that separates the panes).
Use the buttons in the DBRC List History window to initiate other actions:

— To retrieve the most recent information from DBRC, click **Refresh**.

— To toggle between table format and timeline format in the event pane, click **Switch Views**.

— To control the default format, type, and layout of information in the window, click **Preferences**. The List History Preferences dialog box is displayed. For more information, see the online Help topics.

— If you select an eligible data set group, image copy, or change accumulation event, the **DBRC Command** button is active. Click it to display the DBRC Command Interface window (“List History - DBRC Command Interface window” on page 219), from which you can choose to submit one of the displayed commands for the event.

— If you select an eligible data set group, image copy, or change accumulation event, the **GENJCL** button is active. Click it to display the Generate JCL window (“List - History Generate JCL window” on page 220), from which you can generate, edit, save, and submit JCL for the event.

### Events in the DBRC List History Window

The List History window can display information about various types of DBRC events for a selected data set group (DSG) or Fast Path area. The window uses different colors to highlight different types of events.

You can click the **Preferences** button to control the types of events that the window displays. You can click the individual check boxes along the lower edge of a tab to control the types of events that the tab displays.

---

**Note**

The **Logs** check box also controls the display of allocation and deallocation events.

---

### DBRC events

The DBRC List History window supports the following types of events:

- **IC** - image copy of this DSG or area:
  - Batch
  - Concurrent
  - Online
  - SMSNOICIC
—SMSCIC
—SMSOFFLC
—SMSONLC
—User
—User concurrent

- CA - change accumulation run that includes records for this DSG or area
- REORG - reorganization of this DSG or area
- REC - recovery of this DSG or area
- LOG START - open of a Prilog that contains records for this DSG or area
- LOG END - close of a Prilog that contains records for this DSG or area
- ALLOC - allocation of this DSG or area
- DEALLOC - deallocation of this DSG or area
- SWITCH - switch of a log that contains records for this DSG or area

Colors

The window uses colors to represent different types of events as follows:

- Green represents log open, close, and switch events.
- Blue represents DSG or area allocation and deallocation events.
- Orange represents all other events.
List History - DBRC Command Interface window

You can use the List History - DBRC Command Interface window to initiate DBRC commands for the selected object, and monitor the results.

Figure 41: List History - DBRC Command Interface window

<table>
<thead>
<tr>
<th>Object</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data set group (DSG)</td>
<td>■ Set GENMAX value</td>
</tr>
<tr>
<td></td>
<td>■ Set RECOVPD value</td>
</tr>
<tr>
<td></td>
<td>■ Set or reset IC NEEDED flag</td>
</tr>
<tr>
<td></td>
<td>■ Set or reset RECOV NEEDED flag</td>
</tr>
<tr>
<td>Image copy (IC)</td>
<td>■ Set or reset IC1 INVALID status</td>
</tr>
<tr>
<td></td>
<td>■ Set or reset IC2 INVALID status</td>
</tr>
<tr>
<td>Change accumulation (CA) run</td>
<td>Set or reset CARUN INVALID status</td>
</tr>
</tbody>
</table>

The top pane on the left side of the window lists the DBRC commands that are valid for the selected type of object. You can click a command to select it for use.

The following commands are available, depending on the object:

The pane on the top right side of the window displays the information and parameters for the selected command. When you enter the values that you want to use for the parameters and click **Submit**, the interface submits this command for execution.

The bottom pane contains information about the results of the submitted command. The pane logs the DBRC output and the informational and error messages received.
from DBRC. The pane continues to log the output until you close the List History - DBRC Command Interface window.

**Note**

If the DBRCLOG DD statement is present in the Advisor started task, the DBRC List History function creates an audit record of each executed command.

---

**List - History Generate JCL window**

You can use the List History - Generate JCL window to work with solutions for the selected object.

A solution is a predefined set of skeleton JCL and scripts. The interfaces uses a solution to generate executable JCL for the object. For example, the interface uses the IMAGE COPY solution to create JCL that performs an image copy for a data set group.

**Figure 42: List History - Generate JCL window**

The List History - Generate JCL window contains the following tabs:

- Use the Solutions tab to select a solution. Then click **Generate JCL**. The following solutions are available, depending on the object:
Security for the DBRC List History function

You can secure use of the DBRC List History function through the following security access facility (SAF) techniques:

- BMC Advisor-provided security
- IBM DBRC SAF command-level security

**Advisor security**

You implement BMC Advisor-provided security by adding a new general access resource rule:

```
BBM.SDBA.IMSDB.%RECON.NAME%.DBRC
```
In this rule, the `%RECON.NAME%` qualifier is the 8-character name in the
**IMSPLEX/RECON name (used by SAF)** field of the IMSPLEX record. The following
functionality is available, based on the define user access level:

- **NONE**—The DBRC List History function is not available.
- **READ**—The DBRC List History function and the GENJCL button are available.
  The DBRC Commands button is not available.
- **UPDATE**—The complete range of DBRC List History functionality is available.

For more information, see the *Database Products for IMS Customization Guide*.

**DBRC SAF Command-level security**

If the RECON is configured with CMDAUTH=(SAF ...) or CMDAUTH=(BOTH ...),
the DBRC List History function will validate user access to each DBRC command
that users request by using the DBRC Commands button.

The user ID of the Advisor CPC region (subsystem ID xxxxADV) must be authorized
to execute all DBRC queries and commands.

If a user is authorized, the **Submit** button is active in the List History - Generate JCL
window. The user must have a minimum access level of READ in the
BBM.SDBA.IMSDB.#.SUB SAF rule for Advisor. For more information, see the
*Database Products for IMS Customization Guide*.

**Removing a third-party library from the Advisor configuration**

If the STEPLIB concatenation for an IMSPLEX or IMS subsystem includes a library
that contain modules for a third-party product (not from BMC or IBM), and this
library is not included in the STEPLIB concatenation for your batch IMS programs, a
problem might occur in the DBRC List History function.

To resolve the problem, you must remove the third-party library from the Advisor
environment, as explained in the following procedure.

Repeat this procedure for each IMS subsystem that includes the third-party library in
the IMS STEPLIB concatenation.

---

**Note**

This procedure only modifies the Advisor environment; it does not remove the
third-party library from your IMS systems or batch programs.
To remove a third-party library

1. In the console, expand (click) a RECON object to display the associated IMS subsystems.

2. Right-click an IMS subsystem name, and select Maintain IMS Subsystem.

3. Select STEPLIB.

4. Select the third-party library in the concatenation.

5. Click Delete to remove it from the concatenation.

6. Click Save to update the Advisor configuration.
Using the DBRC List History function
Customizing parameters and schedules

This section describes how you can customize parameters and schedules to control Advisor processes.

To control how Advisor processes and utilities work, you can customize parameters as described in “Configuring JCL parameters” on page 225 and “Customizing parameters” on page 226.

To control when various Advisor processes and utilities are executed, you can customize schedules as described in “Customizing schedules” on page 279.

To create and manage dynamic allocation models and matrixes for backup and recovery utilities and processes, you can configure models and matrixes as described in “Configuring dynamic allocation models and matrixes” on page 293.

Configuring JCL parameters

The Configure JCL Parameters wizard provides a step-by-step method for configuring information that Advisor uses when it generates JCL:

- Standard job statements
- Region size limits
- Work file definitions and options
- DBRC usage
- IDCAMS definitions and options
- Definitions of standard DD statements
- Definitions of user variables

Typically, you use the wizard when first implementing Advisor. Therefore, the wizard is available at the IMS Databases level and the IMSPLEX level in the Navigation window of the console. At any time, you can specify JCL parameters at these and lower levels through the Job Control category in the Maintain Parameters window. For more information, see “Job Control parameters” on page 260.
To configure JCL parameters

1. In the Navigation window, right-click the IMS Databases object or an IMSPLEX object and choose Configuration => Configure JCL Parameters.

2. On each page of the wizard, enter the requested information.

   Note
   - Clicking Next or Back navigates from page to page. Clicking Cancel discards your changes and exits.

3. On the last page, click Finish to save your changes and exit.

Customizing parameters

You can control Database Advisor, Recovery Advisor, and other processes by customizing parameters.

Parameters are centrally organized in the console so that you can access, view, and customize all parameters from one window.

Parameter categories

You can customize parameters of the following categories.

Table 11: Parameter categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reorg Thresholds</td>
<td>“Reorg Thresholds parameters” on page 230</td>
</tr>
<tr>
<td>Recovery Thresholds</td>
<td>“Recovery Thresholds parameters” on page 242</td>
</tr>
<tr>
<td>RECON Thresholds</td>
<td>“RECON Thresholds parameters” on page 247</td>
</tr>
<tr>
<td>IMS Status</td>
<td>IMS Status parameters on page 250</td>
</tr>
<tr>
<td>Goals</td>
<td>“Goals parameters” on page 253</td>
</tr>
<tr>
<td>E-mail / WTO / REXX EXEC</td>
<td>“E-mail / WTO / REXX EXEC parameters” on page 253</td>
</tr>
<tr>
<td>IC Triggering by CA</td>
<td>“IC Triggering by CA parameters” on page 256</td>
</tr>
<tr>
<td>Conditional IC</td>
<td>“Conditional IC parameters” on page 257</td>
</tr>
<tr>
<td>Conditional Reorg</td>
<td>“Conditional Reorg parameters” on page 257</td>
</tr>
<tr>
<td>Collect Interval</td>
<td>“Collect interval parameters” on page 258</td>
</tr>
</tbody>
</table>
Levels of parameters

You can customize parameters at the following levels in the object tree:
- Host
- IMSPLEX
- Change accumulation (CA) group
- Database group
- Database
- Partition
- Area
- Data set

For more information about parameters, see the online Help.

Parameter inheritance

Parameter inheritance is a feature that uses the hierarchy of the object tree to facilitate definition of parameters.

Default and customized parameter values

Suggested defaults are provided for most parameters.

The only parameters that do not have suggested defaults are related to site-specific information that you must specify so that optimized JCL can be generated for your environment. The suggested defaults are based on a typical IMS environment.
The suggested defaults are populated at the host level and are inherited by all subordinate objects until you customize parameters at subordinate levels. When you customize a parameter value, all subordinate objects that are specified to inherit the parameter value inherit the new value.

**Note**
A CA group inherits parameter values from the host and IMSPEX levels. A CA group has no subordinate levels.

Most parameters that are customized in the Maintain Parameters window take advantage of parameter inheritance.

**Example**
If you customize a threshold parameter for database group *dbg1*, all objects that are subordinate to database group *dbg1* (databases and data sets) inherit the new threshold parameter. If you also customize the same threshold parameter for database *db3* in database group *

**Database ownership**

Database ownership helps ensure the integrity of parameter values for database objects (and objects that are subordinate to databases).

Although a database can belong to multiple database groups, each database is owned by only one database group. To customize a parameter for a database belonging to more than one database group, you must select the database object which is displayed in the database group that owns the database.

**Example**
Database *db1* was first added to database group *dbg1*. The result is that database group *dbg1* owns database *db1*. Creation of additional database groups resulted in an instance of database *db1* being added to database group *dbg2* and database group *dbg3*. When database group *dbg3* was created, ownership of all databases in database group *dbg3* was transferred to database group *dbg3*. Now database *db1* is owned by database group *dbg3*. To customize any parameters for database *db1*, you must select the instance of database *db1* in database group *dbg3*. To customize parameters for any object that is subordinate to database *db1*, you must select the subordinate object under the database *db1* instance in database group *dbg3*. 
Editing parameters

Use the following procedure to edit parameters at the host level:

1. In the Navigation window, right-click the host object and choose Configuration => Maintain Parameters.

   The Maintain Parameters window (Figure 43 on page 229) is displayed. Parameter categories are displayed on the left side of the window. When you select a parameter category, related parameters are displayed in the middle of the window. Help information is displayed on the right side of the window. To display specific Help information about a parameter (including a description, valid values, and suggested default), click the parameter field name (shown in underlined blue text).

2. Select the parameter category for which you want to customize values, and specify the values.

3. Click Save.
Viewing parameter information

The inherited value and source for a parameter are displayed in the console window in which the parameter is defined.

1. To view details about a parameter, in the Maintain Parameters window click the parameter field name (shown in underlined blue text) for which you want to view details.

On the right side of the window, parameter inheritance information is displayed above the Help information for the selected parameter (see Figure 44 on page 230).

Figure 44: Parameter Detail dialog box

Reorg Thresholds parameters

The following table lists parameters in the Reorg Thresholds category.

Database Advisor uses these parameters during the Analyze process.
### Table 12: Reorg Thresholds parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reorg Thresholds - Default Warning Days category</strong></td>
<td></td>
<td></td>
<td><strong>Warning Days</strong>&lt;br&gt;Specify the number of warning days to use as a default if analyzing the RECON data sets or if analyzing databases and a more specific <strong>Warning Days by Threshold Type</strong> parameter does not apply. If Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Advisor reports a warning (yellow status) to the Exception List.&lt;br&gt; If the Database Availability option is selected as a goal, the suggested default is 45 days.&lt;br&gt; <strong>Example:</strong> Database Advisor projects that a database will be in a dead status on November 11. If the Warning Days threshold is 30, an exception with warning status is reported on the Exception List on October 12.</td>
</tr>
<tr>
<td><strong>Warning Days - Data Set Size</strong></td>
<td>0, 1 to 366</td>
<td>0</td>
<td>Specify the number of warning days (1 through 366) for the following Reorg threshold parameters:&lt;br&gt; ■ VSAM Data Set Size&lt;br&gt; ■ OSAM Data Set Size&lt;br&gt; If Database Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Database Advisor reports a warning (yellow status) to the Exception List.&lt;br&gt; To use the default Warning Days value, specify 0.</td>
</tr>
<tr>
<td><strong>Warning Days - Extents and Volumes</strong></td>
<td>0, 1 to 366</td>
<td>0</td>
<td>Specify the number of warning days (1 through 366) for the following Reorg threshold parameters:&lt;br&gt; ■ VSAM Extents&lt;br&gt; ■ VSAM Volumes&lt;br&gt; ■ OSAM Extents&lt;br&gt; ■ OSAM Volumes&lt;br&gt; If Database Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Database Advisor reports a warning (yellow status) to the Exception List.&lt;br&gt; To use the default Warning Days value, specify 0.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| **Warning Days - Free Space** | 0, 1 to 366     | 0             | Specify the number of warning days (1 through 366) for the following Reorg threshold parameters:  
|                               |                 |               |  ■ Free Blocks %  
|                               |                 |               |  ■ Total Freespace %  
|                               |                 |               |  ■ Alloc Freespace %  
|                               |                 |               |  ■ RAA Free Space %  
|                               |                 |               |  ■ RAA Total Free Space %  
|                               |                 |               |  ■ Total Free Space(LS) %  
|                               |                 |               |  ■ Alloc Free Space(LS) %  
|                               |                 |               | If Database Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Database Advisor reports a warning (yellow status) to the Exception List.  
|                               |                 |               | To use the default Warning Days value, specify 0.                                                                                                                                                                                                                               |
| **Warning Days - Database Performance** | 0, 1 to 366 | 0             | Specify the number of warning days (1 through 366) for the following Reorg threshold parameters:  
|                               |                 |               |  ■ Split Segment %  
|                               |                 |               |  ■ Segments Split to Different Block %  
|                               |                 |               |  ■ I/O Probability Growth %  
|                               |                 |               |  ■ Fragmentation %  
|                               |                 |               |  ■ RAA Smaller by %  
|                               |                 |               |  ■ RAA Larger by %  
|                               |                 |               |  ■ Root Segments in Overflow %  
|                               |                 |               | If Database Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Database Advisor reports a warning (yellow status) to the Exception List.  
|                               |                 |               | To use the default Warning Days value, specify 0.                                                                                                                                                                                                                               |
| **Warning Days - CI & CA Splits** | 0, 1 to 366 | 0             | Specify the number of warning days (1 through 366) for the following Reorg threshold parameters:  
|                               |                 |               |  ■ CI Splits  
|                               |                 |               |  ■ CA Splits  
|                               |                 |               | If Database Advisor forecasts that a statistic will reach or exceed the threshold value within this specified number of days, Database Advisor reports a warning (yellow status) to the Exception List.  
<p>|                               |                 |               | To use the default Warning Days value, specify 0.                                                                                                                                                                                                                               |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI Splits %</td>
<td>0 to 100</td>
<td>20 (indicates 20 percent of total CIs)</td>
<td>Specify the percentage of CI splits at which Database Advisor starts to report potential problems to the Exception List. When the Database Availability option Cannot be taken offline is selected as a goal, this threshold is not active.</td>
</tr>
<tr>
<td>CA Splits %</td>
<td>0 to 100</td>
<td>20 (indicates 20 percent of total CAs)</td>
<td>Specify the percentage of CA splits at which Database Advisor starts to report potential problems to the Exception List. When the Database Availability option Cannot be taken offline is selected as a goal, this threshold is not active.</td>
</tr>
<tr>
<td>Free Blocks %</td>
<td>0 to 100</td>
<td>15 (indicates 15 percent of free blocks)</td>
<td>Specify the percentage of free blocks at which Database Advisor starts to report the INSUFFICIENT FREE BLOCKS exception to the Exception List. Available extents are included in the calculation of free space. This exception is calculated only if the data set is on the last candidate volume.</td>
</tr>
</tbody>
</table>
| Split Segment %           | 0 to 100        | Depends on goal values | Specify the percentage of split segments (number of split segments divided by number of total segments) at which Database Advisor starts to report potential problems to the Exception List. When the specified goal values are selected, the following suggested defaults apply:  
  - 20 (Performance Most Important, indicates 20 percent)  
  - 20 (Performance More Important, indicates 20 percent)  
  - 20 (Equal Importance, indicates 20 percent)  
  - 30 (Space More Important, indicates 30 percent)  
  - 30 (Space Most Important, indicates 30 percent) |
| Segments Split to Different Block % | 0 to 100        | 10            | Specify the percentage of split segments that point to different blocks at which Database Advisor starts to report the TOO MANY SPLITS ACROSS BLOCKS exception to the Exception List. |
I/O Probability Growth %

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| I/O Probability Growth %   | 0 to 10000      | depends on goal values | Specify the percentage of increased I/O probability (calculated as an increase over the Probability of I/O base value) at which Database Advisor starts to report potential problems to the Exception List. If the Probability of I/O base value is less than 10%, Database Advisor uses the value of the Minimum Base I/O Probability % threshold option. When the I/O Probability Growth % threshold is met, an exception is reported on the Exception List, and Database Advisor checks the Improvement % and Fragmentation % thresholds. If both thresholds are met, the exception remains on the Exception List. If one (or neither) threshold is met, the exception is removed from the Exception List, and the I/O Probability Growth % threshold is adjusted. Example: If the Probability of I/O base value for a database is 10 percent after reload and increases to 30 percent over time, the I/O probability growth is 200 percent. If the I/O Probability Growth % threshold is 200 or less, Database Advisor checks the Improvement % and Fragmentation % thresholds to determine whether the exception should remain on the Exception List. When the specified goal settings are selected, the following suggested defaults apply:  
  ■ 50 (Performance Most Important, indicates 50 percent over base value)  
  ■ 50 (Performance More Important, indicates 50 percent over base value)  
  ■ 75 (Equal Importance, indicates 75 percent over base value)  
  ■ 75 (Space More Important, indicates 75 percent over base value)  
  ■ 75 (Space Most Important, indicates 75 percent over base value)  |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improvement %</td>
<td>0 to 100</td>
<td>50 (indicates 50 percent)</td>
<td>Specify the improvement percentage of the number of blocks that a record spans. The Improvement % threshold is checked only if the I/O Probability Growth % threshold is met. The Improvement % threshold works with the Fragmentation % threshold. Example: The I/O Probability Growth % threshold is met, which triggers Database Advisor to analyze the database by using the Improvement % threshold. If the Improvement % threshold is 50, Database Advisor determines how many records in the database can be improved by 50 percent. A record qualifies for a 50 percent improvement if the number of blocks that the record spans can be reduced by half. The number of records that qualify for a 50 percent improvement is used to determine whether the Fragmentation % threshold is met.</td>
</tr>
<tr>
<td>Fragmentation %</td>
<td>0 to 100</td>
<td>depends on goal values</td>
<td>Specify the percentage of the total number of database records that must meet the Improvement % threshold. The Fragmentation % threshold is checked only if the I/O Probability Growth % threshold is met. The Fragmentation % threshold works with the Improvement % threshold. Example: The I/O Probability Growth % threshold is met. Database Advisor analyzes the database by using the Improvement % threshold and determines that 400,000 records (in a database with 1 million records) can be improved by 50 percent. If the Fragmentation % threshold is 40 or less, the exception remains on the Exception List. If the Fragmentation % is 41 or more, the exception is removed from the Exception List and the I/O Probability Growth % threshold is increased. When the specified goal settings are selected, the following suggested defaults apply: ■ 20 (Performance Most Important, indicates 20 percent) ■ 20 (Performance More Important, indicates 20 percent) ■ 30 (Equal Importance, indicates 30 percent) ■ 40 (Space More Important, indicates 40 percent) ■ 40 (Space Most Important, indicates 40 percent)</td>
</tr>
<tr>
<td>Minimum Base I/O Probability %</td>
<td>5 10</td>
<td>10</td>
<td>Specify the minimum base to use in I/O probability growth calculations. Database Advisor calculates the I/O probability growth as an increase over the Probability of I/O base value. If the Probability of I/O base value is less than 10%, Database Advisor uses the Minimum Base I/O Probability % threshold for the calculation.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Total Freespace %         | 0 to 100        | 0             | Specify the percentage of total free space at which Database Advisor starts to report the INSUFFICIENT TOTAL FREE SPACE exception to the Exception List. Available extents are included in the calculation of free space. This threshold is calculated by adding the amount of free space that is available in all of the following areas:  
  ■ Embedded usable free space  
  ■ Available extents  
  ■ High-allocated relative byte address (RBA) minus high-used RBA  
  The purpose of this threshold is to monitor the amount of space that is used for the data set out of the total theoretical maximum amount that is available for the data set. |
| Alloc Freespace %         | 0 to 100        | 0             | Specify the percentage of allocated free space at which Database Advisor starts to report the INSUFFICIENT ALLOCATED FREE SPACE exception to the Exception List. The threshold value is calculated by subtracting the high-used relative byte address (RBA) from the high-allocated RBA and adding the results to the current amount of free space. Available extents are not included in the calculation of free space.  
  The purpose of this threshold is to monitor the amount of usable free space within the current data set allocation. |
| Total Freespace(LS) %     | 0 to 99         | 0             | Specify the percentage of total free space at which Database Advisor starts to report the INSUFF TOTAL FREE SPACE (LS) exception to the Exception List. The purpose of this threshold is to monitor the amount of space that is used for the data set out of the total theoretical maximum amount that is available for the data set. The focus is on whether free space is adequate to handle large segments.  
  Available extents are included in this calculation of free space.  
  This threshold is calculated by adding the amount of free space that is available in all of the following areas:  
  ■ Embedded usable free space that can hold the largest segment size  
  ■ The high-allocated relative byte address (RBA) minus the high-used RBA |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| Alloc Freespace(LS) %     | 0 to 99         | 0             | Specify the percentage of allocated free space at which Database Advisor starts to report the INSUFF ALLOCATED FREE SPACE (LS) exception to the Exception List. The purpose of this threshold is to monitor the amount of usable free space within the current data set calculation. The focus is on whether free space is adequate to handle large segments. The threshold value is calculated by adding the amount of free space that is available in all of the following areas:  
  - Embedded usable free space that can hold the largest segment size  
  - The high-allocated relative byte address (RBA) minus the high-used RBA |
| CI Splits Count           | 0 through 32,767 | 0             | Specify the absolute number of CI splits at which Database Advisor starts to report the TOO MANY CI SPLITS exception to the Exception List. When the Database availability option *Cannot be taken offline* is selected as a goal, this threshold is not active. You can use this threshold in conjunction with the CI Splits % threshold parameter to prevent the exception from being reported for data sets that have a small number of used CIs:  
  - If the number of splits is less than the CI Splits Count threshold, Advisor ignores the CI Splits % threshold.  
  - If the number of splits is equal to or more than the CI Splits Count threshold, Advisor checks the CI Splits % threshold and reports an exception if the percentage is exceeded.  
  To deactivate the threshold, specify 0. If you specify 0, Advisor checks the CI Splits % threshold (if specified) without considering the number of CI splits. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| CA Splits Count            | 0 through 32,767 | 0             | Specify the absolute number of CA splits at which Database Advisor starts to report the TOO MANY CA SPLITS exception to the Exception List. When the Database availability option Cannot be taken offline is selected as a goal, this threshold is not active. You can use this threshold in conjunction with the CA Splits % threshold parameter to prevent the exception from being reported for data sets that have a small number of used CAs:  
  - If the number of splits is less than the CA Splits Count threshold, Advisor ignores the CA Splits % threshold.  
  - If the number of splits is equal to or more than the CA Splits Count threshold, Advisor checks the CA Splits % threshold and reports an exception if the percentage is exceeded.  
  To deactivate the threshold, specify 0. If you specify 0, Advisor checks the CA Splits % threshold (if specified) without considering the number of CA splits. |
| RAA Smaller by %           | 0 to 99         | 75            | Specify the minimum percentage of RAA blocks that are defined in the DBD compared to the value that Database Advisor recommends at which Database Advisor starts to report the HDAM RAA DEFINITION TOO SMALL exception to the Exception List.  
To deactivate the threshold, specify zero. |
| RAA Larger by %            | 0 101 to 999    | 0             | Specify the maximum percentage of RAA blocks that are defined in the DBD, compared to the value that Database Advisor recommends at which Database Advisor starts to report the HDAM RAA DEFINITION TOO LARGE exception to the Exception List.  
To deactivate the threshold, specify zero. |
| RAA Free Space %           | 0 to 100        | 0             | Specify the minimum percentage of usable free space in the RAA at which Database Advisor starts to report the INSUFFICIENT FREE SPACE IN RAA exception to the Exception List.  
To deactivate the threshold, specify zero. |
| RAA Total Free Space %     | 0 to 100        | 0             | Specify the minimum total percentage of free space in the RAA at which Database Advisor starts to report the RAA TOTAL FREE SPACE TOO SMALL exception to the Exception List.  
To deactivate the threshold, specify zero. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Segments in Overflow %</td>
<td>0 to 100</td>
<td>0</td>
<td>Specify the maximum percentage of roots in overflow at which Database Advisor starts to report the TOO MANY ROOTS IN OVERFLOW exception to the Exception List. To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>Honor DBD Free Space Definition %</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether to include the distributed free space that is defined for the RAA in the DBD in Database Advisor calculation for RAA Future Growth. This threshold option works with the RAA Future Growth threshold option to allow Database Advisor to calculate how many RAA blocks you must define in the DBD. This calculation is used to determine whether the RAA Smaller by % threshold and the RAA Larger by % threshold have been met or exceeded.</td>
</tr>
<tr>
<td>Expected RAA Growth %</td>
<td>0 to 100</td>
<td>20</td>
<td>Specify the percentage of future growth that you want Database Advisor to accommodate during the calculation. This threshold option works with the Honor RAA Distribution in Free Space threshold option to allow Database Advisor to calculate how many RAA blocks you must define in the DBD. This calculation is used to determine whether the RAA Larger by % threshold and the RAA Larger by % threshold have been met or exceeded.</td>
</tr>
</tbody>
</table>

**VSAM category of Full Function / HALDB / PDF category of Reorg Thresholds category**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSAM Extents</td>
<td>0 to 253</td>
<td>100</td>
<td>Specify the number of extents at which Database Advisor starts to report potential problems to the Exception List. To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>VSAM Volumes</td>
<td>0 to 32767</td>
<td>8</td>
<td>Specify the number of volumes at which Database Advisor starts to report potential problems to the Exception List. To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>VSAM Data Set Size</td>
<td>0 to 40</td>
<td>35 (indicates 3.5 GB)</td>
<td>Specify the data set size at which Database Advisor starts to report potential problems to the Exception List.</td>
</tr>
<tr>
<td>Required CI Size</td>
<td>0 or 512 to 32768</td>
<td>None</td>
<td>Specify the CI size for the object. If the actual CI size differs from the specified size, Database Advisor reports the INCORRECT CI SIZE exception to the Exception List. The exception has a status of dead (black). Specify a value other than zero only if you want to enforce a defined standard for the object. To deactivate the threshold, specify zero.</td>
</tr>
</tbody>
</table>

**OSAM category of Full Function / HALDB / PDF category of Reorg Thresholds category**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSAM Extents</td>
<td>0 to 120</td>
<td>50</td>
<td>Specify the number of extents at which Database Advisor starts to report potential problems to the Exception List. To deactivate the threshold, specify zero.</td>
</tr>
</tbody>
</table>
### Parameter table

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSAM Volumes</td>
<td>0 to 32767</td>
<td>5</td>
<td>Specify the number of volumes at which Database Advisor starts to report potential problems to the Exception List. To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>OSAM Data Set Size</td>
<td>0 to 80</td>
<td>75 (indicates 7.5 GB)</td>
<td>Specify the data set size at which Database Advisor starts to report potential problems to the Exception List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong> For high availability large database (HALDB) data sets, if the threshold value is greater than 40, Database Advisor uses a default value of 35.</td>
</tr>
</tbody>
</table>

**Fast Path category of Reorg Thresholds category**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOVF Space Utilization %</td>
<td>0 to 100</td>
<td>80 (indicates 80 percent)</td>
<td>Specify the percentage of IOVF control intervals (CIs) to use in the area at which Database Advisor starts to report the AREA IOVF SPACE LOW exception to the Exception List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>SDEP Space Utilization %</td>
<td>0 to 100</td>
<td>80 (indicates 80 percent)</td>
<td>Specify the percentage of SDEP CIs to use in the area at which Database Advisor starts to report the AREA SDEP FREE SPACE TOO LOW exception to the Exception List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>RAA Space Utilization %</td>
<td>0 to 100</td>
<td>0</td>
<td>Specify the percentage of root addressable area (RAA) control intervals at which you want to receive a notification that usage is excessive. When this threshold is crossed, Database Advisor reports the AREA RAA UTILIZATION TOO HIGH exception to the Exception List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To deactivate the threshold, specify 0. BMC deactivates this threshold by default.</td>
</tr>
<tr>
<td>DOVF Space Utilization %</td>
<td>0 to 100</td>
<td>0</td>
<td>Specify the percentage of dependent overflow (DOVF) control intervals at which you want to receive a notification that usage is excessive. When this threshold is crossed, Database Advisor reports the AREA DOVF UTILIZATION TOO HIGH exception to the Exception List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To deactivate the threshold, specify 0. BMC deactivates this threshold by default.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Growth in Record I/O %</td>
<td>0 to 100</td>
<td>80 (indicates 80 percent)</td>
<td>Specify the growth rate percentage at which Database Advisor starts to report the AREA I/O PROBABILITY TOO HIGH exception to the Exception List. To deactivate the threshold, specify zero. The growth rate percentage is the amount of input/output (I/O) which is required for reading the average database record that is used in the area. The amount of I/O which is required for reading the average database record within the area is compared to the amount of I/O which was required at the historical base point. If the current amount exceeds the base amount that is specified by the growth rate threshold, Database Advisor starts to report the HIGH I/O PROBABILITY exception to the Exception List.</td>
</tr>
<tr>
<td>Fragmentation</td>
<td>0 to 1000</td>
<td>10</td>
<td>Specify the fragmentation factor at which Database Advisor starts to report the AREA FRAGMENTATION TOO HIGH exception to the Exception List. To deactivate the threshold, specify zero. The fragmentation factor represents the level of fragmentation or disorganization in the area, based on free space elements and out-of-block pointers.</td>
</tr>
<tr>
<td>Area Used %</td>
<td>0 to 100</td>
<td>None</td>
<td>Specify a threshold percentage of the DEDB area that is used. If the percentage of the DEDB area that is used meets or exceeds this threshold, Advisor reports the AREA USED PERCENT EXCEEDED exception to the Exception List. The exception has a status of dead (black). To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>Reclaimable IOVF for UOW %</td>
<td>0 to 100</td>
<td>None</td>
<td>Specify the percentage of bytes in the IOVF for UOW that could move back into the root area (DOVF) if a UOW reorganization is performed. When this percentage of the minimum IOVF that is reclaimable is exceeded, Advisor reports the RECLAIMABLE IOVF EXCEEDED exception to the Exception List. The exception has a status of dead (black). To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>IOVF UOW's less than total UOW's %</td>
<td>0 to 100</td>
<td>None</td>
<td>Specify the total number of IOVF blocks as a percent of the size of the area (minus SDEP blocks). When the total number of IOVF blocks meets or exceeds this threshold, Advisor reports the TOO FEW IOVF BLOCKS exception to the Exception List. The exception has a status of dead (black). To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>DOVF CI's less than CI's per UOW %</td>
<td>0 to 100</td>
<td>None</td>
<td>Specify the total number of DOVF CIs as a percentage of the size of a unit of work (UOW). If the total number of DOVF CIs meets or exceeds this threshold, Advisor reports the TOO FEW DOVF CONTROL INTERVALS exception to the Exception List. The exception has a status of dead (black). To deactivate the threshold, specify zero.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Required CI Size</td>
<td>0 or 512 to 32768</td>
<td>None</td>
<td>Specify the CI size for the object. If the actual CI size differs from the specified size, Database Advisor reports the INCORRECT CI SIZE exception to the Exception List. The exception has a status of dead (black). Specify a value other than zero only if you want to enforce a defined standard for the object. To deactivate the threshold, specify zero.</td>
</tr>
</tbody>
</table>

**Recovery Thresholds parameters**

This table lists parameters in the Recovery Thresholds category. Recovery Advisor uses these parameters during the analysis process to detect database, change accumulation, and log exceptions. You can set these thresholds at host, IMSplex, and database group levels.

**Table 13: Recovery Thresholds parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Recovery Thresholds category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exclude all objects known to host/RECON/DB Group</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor ignores objects that belong to this host, RECON, or database group (depending on the level at which you are working with parameters). <strong>Note:</strong> If an object belongs to an excluded database group and a group that is not excluded, Recovery Advisor treats the object as excluded. (The excluded group does not have to own the object for this rule to apply.)</td>
</tr>
<tr>
<td>Exclude nonrecoverable objects</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>For applicable exceptions, specify whether Recovery Advisor ignores objects that are defined in the RECON as nonrecoverable.</td>
</tr>
<tr>
<td>Include user image copies</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor considers user image copies as valid when detecting image copy events.</td>
</tr>
</tbody>
</table>

**Recovery Conditions category of the Recovery Thresholds category**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area is not set to preopen</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which a Fast Path area is not set to PREOPEN in the RECON.</td>
</tr>
<tr>
<td>Backout is needed</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an object requires backout (it is in BACKOUT-NEEDED status in the RECON).</td>
</tr>
<tr>
<td>EEQE(s) are detected</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an I/O failure occurred on the media that is storing an object (the object is in EQE FOUND status in the RECON). If a HALDB partition DSG is an object in the group, Recovery Advisor also evaluates and reports events for the associated ILDS and primary index.</td>
</tr>
<tr>
<td>Recovery is needed</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an object requires recovery (it is in RECOV-NEEDED status in the RECON). If a HALDB partition DSG is an object in the group, Recovery Advisor also evaluates and reports events for the associated ILDS and primary index.</td>
</tr>
<tr>
<td>Image copy is needed</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an object requires an image copy (it is in IC-NEEDED status in the RECON).</td>
</tr>
<tr>
<td>Image copy is recommended</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an object should be image copied (it is in IC-RECOMMENDED status in the RECON).</td>
</tr>
<tr>
<td>HALDB partition is not initialized</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is a HALDB that has not been initialized (it is in PARTITION-INIT-NEEDED status in the RECON).</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Object is in read-only status</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is in READ-ONLY status in the RECON.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Object is in prohibit authorization status</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is in PROHIBIT-AUTHORIZATION status in the RECON.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Recovery needed count is invalid</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to check for the condition in which the recovery needed count for an entity (DBD, DEDB, or HALDB partition) does not match</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td>the number of recovery needed flags turned on for the entity’s data set groups (DSGs) or areas.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>No area data set is available</td>
<td>Yes</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which no area data set (ADS) is available for a Fast Path area.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>No area data set is defined</td>
<td>Yes</td>
<td>Yes</td>
<td>Specify whether Recovery Advisor detects the condition in which no ADS has been defined for a Fast Path area.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>Recovery Assets category of the Recovery Thresholds category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No SECLOG entry exists</td>
<td>Yes</td>
<td>Yes, 24 hours, consider batch logs</td>
<td>Specify whether Recovery Advisor detects the condition in which a PRILOG entry does not have a corresponding SECLOG entry within a specified time period or the SECLOG entry is in error. If you specify Yes, also specify the time period and whether to consider batch logs when detecting the condition.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No SECSLD entry exists</td>
<td>Yes</td>
<td>Yes, 24 hours</td>
<td>Specify whether Recovery Advisor detects the condition in which a PRISLD entry does not have a corresponding SECSLD entry within a specified time period or the SECSLD entry is in error. If you specify Yes, also specify the time period.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Assets entry is not cataloged          | Yes  
No  
Inherit      | Yes, 7 days, consider batch logs | Specify whether Recovery Advisor detects the condition in which a recovery asset is not cataloged. Recovery assets include primary and secondary image copies, change accumulations, PRILOG entries, SECLOG entries, PRISLD entries, and SECSLD entries. If you specify Yes, also specify the time period and whether to consider batch logs when detecting the condition. |
| Verify GENMAX value                    | Yes  
No  
Inherit      | No                       | Specify whether Recovery Advisor detects the condition in which the GENMAX value of an object in the RECON is greater than, less than, or not equal to a specified number. If you specify Yes, also specify the GENMAX value that should be used.                                                                                       |
| Verify RECOVPD value                   | Yes  
No  
Inherit      | No                       | Specify whether Recovery Advisor detects the condition in which the RECOVPD value of an object in the RECON is greater than, equal to, or less than a specified number. If you specify Yes, also specify the RECOVPD value that should be used.                                                                 |
| No valid image copy exists             | Yes  
No  
Inherit      | Yes, 7 days              | Specify whether Recovery Advisor detects the condition in which a valid image copy of an object has not completed successfully within a specified time period. This parameter also controls whether Recovery Advisor detects the condition in which no valid image copies are registered in the RECON for an object. If you specify Yes, also specify the time period. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No secondary image copy exists</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which a primary image copy does not have a corresponding valid secondary image copy for an object within the specified time period. If you specify Yes, also specify the time period.</td>
</tr>
<tr>
<td>Number of logs since last IC or CA exceeds a value</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which more than a specified maximum number of logs has been written since the last image copy or change accumulation. In the process of detecting these conditions, Recovery Advisor detects the condition in which no valid image copy exists. If you specify Yes, also specify the maximum number of logs.</td>
</tr>
<tr>
<td>Number of valid image copies is less than required</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>Yes, no limit on the scan</td>
<td>Specify whether Recovery Advisor detects the condition in which a specified number of valid image copies are not registered in the RECON for an object. If you specify Yes, also specify the minimum number of valid image copies that should be registered and whether to limit scanning of the RECON to a specified time period.</td>
</tr>
</tbody>
</table>

**CA/DBDS Groups category of the Recovery Thresholds category**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object is in multiple DBDS groups</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is a member of more than one database data set group (DBDSGRP).</td>
</tr>
<tr>
<td>Object is not in any DBDS group</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is not a member of any DBDSGRP.</td>
</tr>
<tr>
<td>Object is not in any CA group</td>
<td>Yes&lt;br&gt;No&lt;br&gt;Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which an object is not included in a change accumulation group.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Verify GRPMAX value</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which the change accumulation group GRPMAX value is greater than, less than, or not equal to a specified number. If you specify Yes, also specify the GRPMAX value that should be used.</td>
</tr>
<tr>
<td>No CA run exists</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which a CAGRP does not have a CA run within a specified time period or the CA run is in error. If you specify Yes, also specify the time period.</td>
</tr>
<tr>
<td>Number of CA runs is less than</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether Recovery Advisor detects the condition in which the specified number of CA run records are not registered in the RECON for a CAGRP. If you specify Yes, also specify the minimum number of valid CA run records that should be registered and whether to limit the scan of the RECON to a specified time period.</td>
</tr>
</tbody>
</table>

**RECON Thresholds parameters**

Recovery Advisor uses RECON Thresholds parameters during analysis to detect RECON data set exceptions.

The RECON Reorg utility uses RECON Operational Controls parameters during execution to resolve RECON contention and request RECON backups. You can set these thresholds at host and IMSPLEX levels.

For RECON thresholds, the following general rules apply:

- To inactivate the threshold, specify 0 for the value.
- To use the inherited value, clear the value.

*Table 14 on page 248 lists parameters in the RECON Thresholds category.*
### Table 14: RECON Thresholds parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RECON Thresholds category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CI Splits %</td>
<td>0 to 99</td>
<td>20 (indicates 20 percent of total CIs)</td>
<td>Specify whether Recovery Advisor detects that a RECON data set has exceeded the specified percentage of control interval (CI) splits, relative to the total number of CIs in the data set.</td>
</tr>
<tr>
<td>CA Splits %</td>
<td>0 to 99</td>
<td>20 (indicates 20 percent of total CAs)</td>
<td>Specify whether Recovery Advisor detects that a RECON data set has exceeded the specified percentage of control area (CA) splits, relative to the total number of CAs in the data set.</td>
</tr>
<tr>
<td>Alloc Freespace %</td>
<td>0 to 99</td>
<td>15 (indicates 15 percent of free space)</td>
<td>Specify whether Recovery Advisor detects that a RECON data set has less than the specified percentage of free space, relative to the allocated space in the data set.</td>
</tr>
<tr>
<td>Extents</td>
<td>0 to 253</td>
<td>1</td>
<td>Specify whether Recovery Advisor detects that a RECON data set has exceeded the specified number of extents.</td>
</tr>
<tr>
<td>Volumes</td>
<td>0 to 58</td>
<td>1</td>
<td>Specify whether Recovery Advisor detects that a RECON data set has exceeded the specified number of volumes.</td>
</tr>
<tr>
<td>Data set size (tracks)</td>
<td>0 to 9999</td>
<td>0 (indicates that this threshold is deactivated)</td>
<td>Specify whether Recovery Advisor detects that the number of tracks for a RECON data set has exceeded the specified number of tracks.</td>
</tr>
<tr>
<td>PRILOG Record Size %</td>
<td>0</td>
<td>0</td>
<td>Specify whether to check for the condition in which an open PRILOG has exceeded the specified percentage (1 through 99) of the maximum 16-MB record size. Specify 0 to ignore this condition. Specify a null value to use the inherited value.</td>
</tr>
</tbody>
</table>

Customizing parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON Warning Days</td>
<td>1 through 366</td>
<td>Null</td>
<td>Specify the time period (in days, 1 through 366) that Recovery Advisor uses in calculations that forecast whether a RECON-related statistic will reach or exceed a threshold value within that time period. (If a statistic is forecast to reach or exceed the threshold, Recovery Advisor issues a warning-level exception.) Specify a null value to use the inherited value.</td>
</tr>
<tr>
<td>RECON Warning Days</td>
<td>Null</td>
<td>45</td>
<td>Specifying the time period (in days, 1 through 366) that Recovery Advisor uses in calculations that forecast whether a RECON-related statistic will reach or exceed a threshold value within that time period. (If a statistic is forecast to reach or exceed the threshold, Recovery Advisor issues a warning-level exception.) Specify a null value to use the inherited value.</td>
</tr>
<tr>
<td>RECON Detail Record Retention (days)</td>
<td>1 through 32767</td>
<td>Null</td>
<td>Specify the maximum number of days (1 through 32767) that Recovery Advisor should retain a PRILOG status record for an open, online IMS system that is defined to Recovery Advisor. Recovery Advisor automatically deletes records that are older than this number of days. Specify a null value to use the inherited value.</td>
</tr>
<tr>
<td>RECON Data Set Contention - Resolution option</td>
<td>Depend on RECON Loss Notification WTOR to notify operators RMGR to notify IMS subsystems DELTA PLUS to notify IMS subsystems Fail utility with return code Fail utility with abend code Inherit</td>
<td>Fail utility with return code</td>
<td>Specify the action to take if the RECON Reorg utility encounters a RECON contention problem. If you select the <strong>WTOR to notify operators</strong> option, also specify the text of the WTOR message to be issued. If you select the <strong>Fail utility with abend code</strong> option, also specify the abend code to be issued.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RECON Data Set Contention - Retry Time Period (Minutes)</td>
<td>Blank, 0, 10, 20, 30, 40, 50, 60</td>
<td>20</td>
<td>Specify the maximum amount of time (in minutes) that the RECON Reorg utility should wait or attempt to resolve RECON contention problems. If you do not want the utility to wait if RECON contention occurs, set the value to 0. To use the inherited value, set the parameter to blank.</td>
</tr>
<tr>
<td>RECON Data Set Contention - Return Code</td>
<td>0 to 99</td>
<td>12</td>
<td>Specify the return code that the RECON Reorg utility issues if a RECON contention problem is not resolved. To use the inherited value, clear the parameter value.</td>
</tr>
<tr>
<td>RECON Backup Data Set Name</td>
<td>Data set name(1 to 44 alphanumeric characters)</td>
<td>none</td>
<td>Specify the data set name to use to back up the RECON data sets before the RECON Reorg utility performs any functions. To use the inherited value, clear the parameter value. If the inherited value is blank, no RECON backup is performed.</td>
</tr>
</tbody>
</table>

### IMS Status parameters

The IMS Status Monitoring feature uses the parameters in the IMS Status category to determine whether and how to detect and report exceptional status conditions for IMS online systems.

**Tip**

For more information, view the Quick Course "MAXM Database Advisor - Monitoring IMS Status."

Each parameter consists of the following specifications:

- Whether to monitor a specific type of condition
  
  For most parameters, this specification is Yes or No; for a few parameters, this specification is a numeric setting.
- If the condition will be monitored, when to begin monitoring and when to end

The Begin and End specifications accommodate time-frames (such as maintenance windows) when you expect a condition to occur but do not want Advisor to report that as an exception.

The following parameters are included in this category:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Online Regions Not Avail</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which an IMS online region is inactive.</td>
</tr>
<tr>
<td>IMS Not Available</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which an IMS control region is inactive on all systems in which Advisor is running.</td>
</tr>
<tr>
<td>HMS Recall Pending</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a hierarchical storage management (HSM) recall has been issued for a database.</td>
</tr>
<tr>
<td>DBD/Area Not Open</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a DBD or area is closed.</td>
</tr>
</tbody>
</table>
| SDEP Freeblks Low %        | 1 through 99 0 (ignore this condition) | 0             | Specify whether to detect the condition in which the specified percentage of freespace in the SDEP section is not available for use, according to the DMAC.  
**Note:** In a data sharing environment, the copy of DMAC in the control region might be out of sync with DASD. Also the area must be open for Advisor to obtain the value. |
<p>| DBD/Area Locked            | Yes, No, Inherit | No            | Specify whether to detect the condition in which a DBD or area is in locked status. |
| DBD/Area Init Fail         | Yes, No, Inherit | No            | Specify whether to detect the condition in which a DBD or area is in a non-INIT status. |
| DBD Deleted by OLCHNG      | Yes, No, Inherit | No            | Specify whether to detect the condition in which a DBD was removed by the online change process. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alloc Failed</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which an attempt to allocate a database was unsuccessful.</td>
</tr>
<tr>
<td>Too Many RDWRT Fails</td>
<td>1 through 99 0 (ignore this condition)</td>
<td>0</td>
<td>Specify whether to detect the condition in which the specified total number of read and write failures has occurred or has been exceeded.</td>
</tr>
<tr>
<td>STA Cmd Pending</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a STA command has been issued for a database, but the command has not yet completed.</td>
</tr>
<tr>
<td>DBR Cmd Pending</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a DBR command has been issued for a database, but the command has not yet completed.</td>
</tr>
<tr>
<td>DBD Cmd Pending</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a DBD command has been issued for a database, but the command has not yet completed.</td>
</tr>
<tr>
<td>DBD/Area Stopped</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a DBD or area is in a STOPPED status.</td>
</tr>
<tr>
<td>DBRC Auth Fail</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which an authorization call to DBRC has failed.</td>
</tr>
<tr>
<td>Read Errors</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a READ EQE count is greater than 0.</td>
</tr>
<tr>
<td>Write Errors</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a WRITE EQE count is greater than 0.</td>
</tr>
<tr>
<td>DBD In Error</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which a backout failure or other error condition has occurred for a database.</td>
</tr>
<tr>
<td>Area Stopped</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to detect the condition in which an area is in STOPPED status.</td>
</tr>
</tbody>
</table>
### Goals parameters

This table lists parameters in the Goals category.

The Solution process uses these parameters to recommend solutions.

**Table 15: Goals parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space vs. Performance Weight Factor</td>
<td>Space Most Important</td>
<td>Equal Importance</td>
<td>Specify whether space or performance is more important. When you change a goal, the I/O Probability Growth threshold is changed to the default for the selected goal.</td>
</tr>
<tr>
<td></td>
<td>Space More Important</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Equal Importance</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Performance More Important</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Performance Most Important</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Availability</td>
<td>Cannot be taken offline</td>
<td>Can take offline periodically</td>
<td>Specify database availability requirements.</td>
</tr>
<tr>
<td></td>
<td>Can take offline briefly</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Can take offline periodically</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Can take offline when necessary</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherited</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### E-mail / WTO / REXX EXEC parameters

This table lists parameters in the E-mail / WTO / REXX EXEC category.

Advisor uses these parameters to send notifications about detected exceptions or to generate REXX EXEC batch jobs.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mail / WTO category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>From</td>
<td>1 to 70 alphanumeric characters</td>
<td>None</td>
<td>Specify the sender of a notification. This value must be a valid e-mail address of the form name@domain. While the From e-mail address need not be a working e-mail address, the specified address must contain no syntax errors. You can provide a parenthetical text string after the e-mail address. For recipients, this string can further clarify the source of the message.</td>
</tr>
<tr>
<td>Subject</td>
<td>1 to 70 alphanumeric characters</td>
<td>None</td>
<td>Specify the subject line text for an e-mail notification. You can use this field to identify the component that triggered the e-mail alert.</td>
</tr>
<tr>
<td>Issue WTO’s</td>
<td>Selected</td>
<td>Not selected</td>
<td>Not selected</td>
</tr>
<tr>
<td></td>
<td>Not selected</td>
<td></td>
<td>Specify whether to send a write-to-operator (WTO) message each time an e-mail alert is generated. To issue WTO messages with e-mail alerts, select the Send WTO check box. To issue WTO messages only (no e-mail alerts), you can divert e-mail alerts to an invalid e-mail address. WTO messages are still issued.</td>
</tr>
<tr>
<td>Recovery Failures Email Notification</td>
<td>Yes</td>
<td>Inherit</td>
<td>Specify a value for this parameter to control whether the Monitor Recovery function generates a notification message when a database recovery fails.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTML Format</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>Inherit</td>
<td>Specify whether to use HTML format for e-mail notifications (Yes) or plain-text format (No).</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Consolidate E-mail messages</td>
<td>None</td>
<td>None</td>
<td>At the IMSPLEX level, this parameter provides a shortcut to a dialog box for editing the E-MAIL CONSOLIDATION schedule. At lower levels, this parameter provides information about whether the schedule is enabled or disabled.</td>
</tr>
<tr>
<td>Invoke REXX EXEC</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether a REXX EXEC is invoked when an exception is detected.</td>
</tr>
<tr>
<td>REXX EXEC Name</td>
<td>REXXEXECMemberName, Inherit</td>
<td>None</td>
<td>Specify the REXX EXEC member name containing the REXX EXEC to be invoked when an exception is detected. To use the inherited value for this parameter, specify Inherit. If you do not specify a value, REXX EXEC processing does not occur.</td>
</tr>
<tr>
<td>REXX EXEC Library</td>
<td>REXXEXECDataSetName, Inherit</td>
<td>None</td>
<td>Specify the REXX EXEC library data set name containing the REXX EXEC to be invoked when an exception is detected. To use the inherited value for this parameter, specify Inherit. If you do not specify a value, REXX EXEC processing does not occur.</td>
</tr>
<tr>
<td>Defer REXX EXEC until scheduled</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to defer REXX EXEC batch job execution according to a schedule.</td>
</tr>
</tbody>
</table>

**Warning, Critical, and Dead categories of the E-Mail / WTO / REXX EXEC category**

<table>
<thead>
<tr>
<th>Category</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warning</td>
<td>1 to 70 alphanumeric characters</td>
<td>None</td>
<td>Specify e-mail addresses to which warning status notifications are sent.</td>
</tr>
<tr>
<td>Critical</td>
<td>1 to 70 alphanumeric characters</td>
<td>None</td>
<td>Specify e-mail addresses to which critical status notifications are sent.</td>
</tr>
<tr>
<td>Dead</td>
<td>1 to 70 alphanumeric characters</td>
<td>None</td>
<td>Specify e-mail addresses to which dead status notifications are sent.</td>
</tr>
</tbody>
</table>
### IC Triggering by CA parameters

The following table lists parameters in the IC Triggering by CA category.

Use these parameters to control the Image Copy (IC) Triggering by Change Accumulation (CA) feature. For more information, see “Using the IC Triggering by CA feature” on page 385.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze CA Group to Trigger ICs</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to enable the IC Triggering by CA feature. This parameter causes the feature to analyze the size of the CA group during CA processing.</td>
</tr>
<tr>
<td>CA File Size Threshold</td>
<td>1 through 1024 KB, MB, GB, or TB</td>
<td>100 GB</td>
<td>Specify a threshold value for triggering IC processing for the CA group.</td>
</tr>
<tr>
<td>When CA file size threshold is exceeded</td>
<td>Select top &quot;nn&quot; members of CA group for IC after ranking by size Select top &quot;nn&quot; members of CA group for IC after ranking by # records Select any member of CA group for IC with more than &quot;nn&quot; bytes Inherit Value</td>
<td>Select top &quot;nn&quot; members of CA group for IC after ranking by size</td>
<td>Specify how to select members of the CA group for image copying if the size of the CA data set exceeds the value that is specified for the CA File Size Threshold parameter</td>
</tr>
<tr>
<td>Select any member of CA group for IC if &quot;nn&quot; days since last IC</td>
<td>1 through 99</td>
<td>7</td>
<td>Specify whether to trigger an IC of any member of the CA group by the age of the last IC for that member.</td>
</tr>
</tbody>
</table>
Conditional IC parameters

The following table lists parameters in the Conditional IC category.

The Conditional Image Copy feature uses these parameters during processing. For more information, see “Using the Conditional Image Copy feature” on page 390.

Table 18: Conditional IC parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skip IC if no updates</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to activate the Conditional Image Copy feature for the selected object (host, IMSPLEX, database group, database, partition, or area).</td>
</tr>
<tr>
<td>Perform IC if &quot;nn&quot; days since last IC</td>
<td>1 to 99</td>
<td>7</td>
<td>Specify the maximum number of days that can elapse before the Conditional Image Copy feature allows a scheduled batch image copy to be performed, regardless of whether database updates have occurred since the last image copy was performed. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

Conditional Reorg parameters

The following table lists parameters in the Conditional Reorg category.

The Conditional Reorg feature uses these parameters during processing. The Conditional Reorg feature applies to full-function databases and high availability large databases (HALDBs) only; it does not apply to Fast Path areas.
Table 19: Conditional Reorg parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activate Conditional Reorg</td>
<td>Yes, No, Simulated, None</td>
<td>Simulated</td>
<td>Specify whether to activate the Conditional Reorg feature.</td>
</tr>
<tr>
<td>Allow Reorg</td>
<td>0 to 365</td>
<td>45</td>
<td>Specify the number of days before the predicted error date at which Database Advisor lets reorganization run, regardless of conditional reorganization results. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Completion Code Type</td>
<td>Abend, RC (reason code), None</td>
<td>RC</td>
<td>Specify the type of completion code to return for conditional reorganization jobs.</td>
</tr>
<tr>
<td>Completion Code</td>
<td>valid abend code values: 4000 to 4094, valid return code values: 00 to 99 (except 16)</td>
<td>3 (for RC completion code type)</td>
<td>Specify a numeric value for abend and return code types. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

Collect interval parameters

The following table lists parameters in the Collect Interval category.

Database Advisor uses these parameters for the Collect process.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Collection Interval</td>
<td>0 to 365</td>
<td>45</td>
<td>Specify the frequency (in days) to collect statistics for the selected object. The frequency interval is checked by the FF COLLECT function or the FP COLLECT function to determine whether statistics should be gathered so that every database, partition, or area object would not be processed each time. If the difference between the last day that statistics were gathered and the current day is less than the specified interval, the object is skipped. This parameter is active only if the FF COLLECT function or the FP COLLECT function is scheduled to run automatically. If you do not want to collect data for the selected object, specify zero. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Activate Fast Path Space Data Collection</td>
<td>Inherit Yes No</td>
<td>No</td>
<td>Specify whether the FP SPACE COLLECT function should run at the scheduled interval (in hours) to gather IOVF and SDEP statistics for a Fast Path area data set. This parameter is active only if the FP SPACE COLLECT function is scheduled to run automatically. <strong>WARNING:</strong> Because of the potential for high resource consumption when the FP SPACE COLLECT function is executed, you should <em>not</em> specify Yes for this option at the RECON level or a higher level. You should specify Yes for this option only at the database group level or area level.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Activate FF Hourly Collect</td>
<td>Inherit, Yes, No</td>
<td>No</td>
<td>Specify whether the FF COLLECT HOURLY function should run at the scheduled interval (in hours) to gather statistics for full-function databases and HALDBs. This parameter is active when the FF COLLECT HOURLY function is scheduled to run automatically. <strong>WARNING:</strong> Because of the potential for high resource consumption when the FF COLLECT HOURLY function is executed every hour, you should not specify Yes for this option at the RECON level or higher. Specify Yes only at the database level or lower.</td>
</tr>
</tbody>
</table>

**Analysis parameters**

The following table lists parameters in the Analyze category.

Database Advisor uses these parameters during the Analyze process.

**Table 21: Analysis parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activate Analysis</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether to activate the Database Advisor analysis process.</td>
</tr>
<tr>
<td>Run Analysis After IBM HD Load</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether to activate the Database Advisor analysis process after IBM loads.</td>
</tr>
</tbody>
</table>

**Job Control parameters**

The following table lists parameters in the Job Control category. The Generate JCL function uses these parameters to create JCL.
Table 22: Job Control parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Control category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Job Card 1</td>
<td>1 to 72 alphanumeric characters</td>
<td>None</td>
<td>Specify a z/OS JOB card statement.</td>
</tr>
<tr>
<td>Job Card 2, 3, 4</td>
<td>1 to 72 alphanumeric characters (per box)</td>
<td>None</td>
<td>Continue a z/OS job card statement from the previous text box. If no continuation is required, you must specify //** to indicate a comment line.</td>
</tr>
<tr>
<td>Region category of the Job Control category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Small Region</td>
<td>0 to 9999</td>
<td>1024K</td>
<td>Specify the smallest region size. Use the format nnnnK or nnnnM. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Medium Region</td>
<td>0 to 9999</td>
<td>4096K</td>
<td>Specify the midrange region size. Use the format nnnnK or nnnnM. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Large Region</td>
<td>0 to 9999</td>
<td>0M</td>
<td>Specify the largest region size. Use the format nnnnK or nnnnM. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SYSOUT Class</td>
<td>A to Z, 0 to 9, or *</td>
<td>*</td>
<td>Specify an output class to use on SYSOUT DD statements in generated JCL. An asterisk or a dollar symbol indicates that the MSGCLASS parameter on the JOB statement is used for the SYSOUT class DD statement; all job output goes to the same output class. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Work File UNIT Name</td>
<td>1 to 8 alphanumeric characters</td>
<td>WORK</td>
<td>Specify the JCL UNIT parameter to use for work file DD statements in generated JCL. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Work File High-Level Qualifier</td>
<td>1 to 16 alphanumeric characters</td>
<td>WORK.HILVL.QUAL</td>
<td>Specify the high-level qualifier to use for work file data sets in generated JCL. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Retention Period</td>
<td>0 to 9999</td>
<td>90</td>
<td>Specify the number of days to retain the work file, unload file, or image copy tape. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Include SMS Classes in JCL</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to use SMS. <strong>Note:</strong> The IDCAMS(MODEL) keyword is generated only if you set the value to Yes.</td>
</tr>
<tr>
<td>SMS Data Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the data class to use to obtain information for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SMS Management Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the management class to use to obtain data management information (migration, backup, and retention criteria) for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SMS Storage Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the storage class to use to obtain storage information for the data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Use SMS for Work Files</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to let SMS manage work file storage.</td>
</tr>
<tr>
<td>Sort Work UNIT Name</td>
<td>1 to 8 alphanumeric characters</td>
<td>WORK</td>
<td>Specify the JCL UNIT parameter to use for Sort work file DD statements in generated JCL. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Space Allocation Type</td>
<td>CYL (cylinders)</td>
<td>CYL</td>
<td>Specify the method of allocating space.</td>
</tr>
<tr>
<td>IDCAMS category of the Job Control category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDCAMS Delete Control Statements</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set that contains IDCAMS control statements to delete databases. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>IDCAMS Define Control Statements</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set that contains IDCAMS control statements to define databases. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Full Function Member Type</td>
<td>Database</td>
<td>Database</td>
<td>Specify the means of storing IDCAMS delete and define control statements in the delete and define data set.</td>
</tr>
</tbody>
</table>
### Unload parameters

The following table lists parameters in the Unload category. The Generate JCL function uses these parameters to create database unload JCL.

**Table 23: Unload parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unload category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Tape</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to write the unload file to tape or DASD.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UNIT Name</td>
<td>1 to 8 alphanumeric characters</td>
<td></td>
<td>Specify the type of device on which the unload data set resides. You can specify a group of devices. For valid names, consult your systems programmer. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Retention Period</td>
<td>0 to 999</td>
<td>90</td>
<td>Specify the number of days to retain the Unload file. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Include SMS Classes in JCL</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to use SMS.</td>
</tr>
<tr>
<td>SMS Data Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the data class to use to obtain information for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SMS Management Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the management class to use to obtain data management information (migration, backup, and retention criteria) for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SMS Storage Class</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the storage class to use to obtain storage information for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

**Fast Path category of the Unload category**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unload Data Set Mask Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the unload data set mask name. The name is used to catalog the output file from the Fast Path Unload function and is used as the input file for the Fast Path Reload function. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Expand Unload</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to write out the unload file expanded or compressed.</td>
</tr>
</tbody>
</table>

**Full-Function/HALDB / PDF category of the Unload category**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unload Data Set Name Prefix</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the prefix to use for the unload data set. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

**Reorg parameters**

The following table lists parameters in the Reorg category.

The Generate JCL function uses these parameters to create database reorganization JCL.

**Table 24: Reorg parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast Path category of the Reorg category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reorg Options</td>
<td>IOVF</td>
<td>IOVF</td>
<td>Specify the option for selecting UOWs to be reorganized by the Fast Path Reorg function.</td>
</tr>
<tr>
<td>IOVF Threshold Type</td>
<td>Percent Number Inherit</td>
<td>Inherit</td>
<td>Specify the type of threshold value to use for specifying the minimum amount of UOWs of IOVF that must be saved. If the minimum amount of saved UOWs is not met, the Fast Path Reorg function bypasses the reorganization of the UOW. If this parameter is selected, a value must be specified for the <strong>IOVF Threshold Value</strong> field.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------------------------------------------------------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IOVF Threshold Value</td>
<td>percent values 1 to 100; whole numbers 1 to 32767</td>
<td>None</td>
<td>Specify the minimum amount of UOWs of IOVF that must be saved before the Fast Path Reorg function reorganizes the UOW. IOVF Threshold Type of percentage or number must also be specified. The threshold value that you specify must be appropriate for the IOVF Threshold Type selection. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Full Function / HALDB / PDF category of the Reorg category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMSOUT Data Set Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify a data set name that contains IDCAMS ALTER commands to be used during the swapping phase of a MAXM Reorg function.</td>
</tr>
<tr>
<td>Image Copy Indexes</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to require image copies for indexes.</td>
</tr>
<tr>
<td>Build Unload During Reorg</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to build an unload file during reorganization.</td>
</tr>
<tr>
<td>Log Data Set Name Prefix</td>
<td>1 to 18 alphanumeric characters</td>
<td>None</td>
<td>Specify the high-level qualifier for the IEFRDER and CRFRDER data set names. IEFRDER and CRFRDER are created during execution of the Online Reorg function. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Logging</td>
<td>Single</td>
<td>Single</td>
<td>Specify whether to create one or two copies of the log data set.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Log to Tape</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to write the log data set to tape.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Log UNIT</td>
<td>1 to 8 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set name to use when allocating log data sets.</td>
</tr>
<tr>
<td>Shadow Delete Control</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set that contains IDCAMS control statements to delete shadow databases. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Statements</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shadow Define Control</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set that contains the IDCAMS control statements to define shadow databases. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Statements</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shadow Data Set Suffix</td>
<td>1 to 3 alphabetic characters</td>
<td>Z</td>
<td>Specify a suffix to append to the original database data sets when executing a MAXM Reorg function that uses shadow databases. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

### Image Copy parameters

The following table lists parameters in the Image Copy category. The Generate JCL function uses these parameters to create image copy JCL.

**Table 25: Image Copy parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Copy category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Number of Copies</td>
<td>Single</td>
<td>Single</td>
<td>Specify whether to create one or two copies of the image copy data set.</td>
</tr>
<tr>
<td></td>
<td>Dual</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Processing Mode</td>
<td>ASYNC</td>
<td>ASYNC</td>
<td>Specify whether to invoke image copy processing in asynchronous mode (by using the AIC command) or in synchronous mode (by using the IC command).</td>
</tr>
<tr>
<td></td>
<td>SYNC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICP Global Options</td>
<td>1 to 8 alphanumeric</td>
<td>None</td>
<td>Specify the name of the global options module that contains the option values to use during execution of the BMC Image Copy utility. In generated JCL for the utility, this name is used as the value of the PARMLIB keyword.</td>
</tr>
<tr>
<td></td>
<td>characters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Set Name Prefix</td>
<td>1 to 8 alphanumeric</td>
<td>None</td>
<td>Specify a name to use as a prefix when constructing the data set name to dynamically allocate image copy data sets. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td></td>
<td>characters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use Tape</td>
<td>Yes</td>
<td>No</td>
<td>Specify whether to write the file to tape or DASD. If the Use Tape parameter and the Use SMS parameter are set to Yes, the image copy is written to tape.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inherit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT Name</td>
<td>1 to 8 alphanumeric</td>
<td>None</td>
<td>Specify the type of device on which the image copy data set resides. You can specify a group of devices. For valid names, consult your systems programmer. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td></td>
<td>characters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Retention Period</td>
<td>0 to 999</td>
<td>90</td>
<td>Specify the number of days to retain the work file, the unload file, or the image copy tape. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Use SMS</td>
<td>Yes/No/Inherit</td>
<td>No</td>
<td>Specify whether to use SMS. If the Use Tape parameter and the Use SMS parameter are set to Yes, the image copy is written to tape.</td>
</tr>
<tr>
<td>SMS Data Class</td>
<td>8 alphanumeric</td>
<td>None</td>
<td>Specify the data class to use to obtain information for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SMS Management Class</td>
<td>8 alphanumeric</td>
<td>None</td>
<td>Specify the management class to use to obtain data management information (migration, backup, and retention criteria) for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>SMS Storage Class</td>
<td>8 alphanumeric</td>
<td>None</td>
<td>Specify the storage class to use to obtain storage information for data set allocation. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>ICP Image Copy Model</td>
<td>8 alphanumeric</td>
<td>None</td>
<td>Specify the name of the predefined dynamic allocation model to use for allocating output image copy data sets during execution of the BMC Image Copy utility.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>GDG</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to allocate the image copy data set as a generation data group (GDG).</td>
</tr>
<tr>
<td><strong>Fast Path category of the Image Copy category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output Image Copy Mask Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the output image copy data set mask name that is cataloged from Fast Path Reorg, Unload, Change, Extend, and Image Copy functions. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Dual Output Image Copy Mask Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the dual output image copy data set mask name that is cataloged from Fast Path Reorg, Unload, Change, Extend, and Image Copy functions. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Notify DBRC of Image Copy</td>
<td>Yes, No, Inherit</td>
<td>Yes</td>
<td>Specify whether to notify DBRC of image copy creation.</td>
</tr>
<tr>
<td>Compression Technique</td>
<td>FSE, CCC, DPE</td>
<td>No</td>
<td>Specify the type of compression to use for compression of image copies. To use the inherited value, select the blank item.</td>
</tr>
<tr>
<td><strong>Full Function / HALDB / PDF category of the Image Copy category</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Image Copy</td>
<td>Yes, No, Inherit</td>
<td>No</td>
<td>Specify whether to create virtual image copies.</td>
</tr>
</tbody>
</table>
### Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Data Set Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the data set name that is recorded in DBRC for the image copy of an index. The data set name does not physically exist. To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

#### Snapshot category of the Image Copy category

<table>
<thead>
<tr>
<th>Snapshot Option</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional IC</td>
<td>Traditional IC</td>
<td>None</td>
<td>Specify the type of image copy to create. This parameter controls whether generated JCL for the BMC Image Copy utility contains keywords that enable and control Snapshot Copy and Instant Snapshot processing. This processing is obtained through the EXTENDED BUFFER MANAGER (XBM) for IMS product or the SNAPSHOT UPGRADE FEATURE for IMS component.</td>
</tr>
<tr>
<td>Snapshot Copy Only</td>
<td>Traditional IC</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Snapshot or Online IC</td>
<td>Traditional IC</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Instant Snapshot Only</td>
<td>Traditional IC</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Instant Snapshot or Traditional IC</td>
<td>Traditional IC</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Inherit</td>
<td>Traditional IC</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>XBM Subsystem ID</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 to 8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the XBM subsystem to use for Snapshot Copy and Instant Snapshot processing.</td>
</tr>
</tbody>
</table>

#### Additional Fast Path parameters

The following table lists parameters in the Additional Fast Path category. The Generate JCL function uses these parameters to create JCL for Fast Path functions.
Table 26: Additional Fast Path parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Image Copy Mask Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the input image copy data set mask name that is used for Fast Path Unload, Change, Extract, and Analyzer functions. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Extract Data Set Mask Name</td>
<td>1 to 44 alphanumeric characters</td>
<td>None</td>
<td>Specify the extract data set mask name that is cataloged from the Fast Path Extract function. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Pointer Validation Type</td>
<td>Quick, Full, Off, No</td>
<td>Quick</td>
<td>Specify pointer validation methods to use with the Analyzer function.</td>
</tr>
<tr>
<td>SDEP Pointer Validation Type</td>
<td>Quick, Full, Off, No</td>
<td>Quick</td>
<td>Specify the SDEP pointer validation methods to use with the Analyzer function. To use the inherited value, select the blank item.</td>
</tr>
<tr>
<td>Invoke Compression</td>
<td>Reload Command, Change Command, Both, No</td>
<td>No</td>
<td>Specify the compression option to compress data during Change or Reload functions. To use the inherited value, select the blank item.</td>
</tr>
</tbody>
</table>

Repository Management parameters

The following table lists parameters in the Repository Management category. Advisor functions use these parameters to manage data in repositories.
Table 27: Repository Management parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repository management category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capture Statistics</td>
<td>Yes, No, None</td>
<td>Yes</td>
<td>Specify whether to capture statistics for a selected object. To use the inherited value, select <strong>None</strong>. Statistical information is gathered for the selected object only if statistical information is gathered for its parent object.</td>
</tr>
<tr>
<td>Detail Record Retention (days)</td>
<td>1 to 32767</td>
<td>32</td>
<td>Specify the maximum number of days to retain a detail statistical record. When detail records have been retained for the specified number of days, they are deleted even if the maximum number of detail records to retain has not been reached. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Maximum Stats Detail Record</td>
<td>0 to 32767</td>
<td>10</td>
<td>Specify the maximum number of detail statistical record instances to retain. A detail record instance is captured after every utility execution. <strong>WARNING:</strong> Warning: If you specify zero, all detail records are purged from the repository when the PURGE function executes. To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>--------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Maximum Monthly Records</td>
<td>1 to 32767</td>
<td>25</td>
<td>Specify the maximum number of monthly statistical records to retain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Maximum Yearly Records</td>
<td>1 to 32767</td>
<td>7</td>
<td>Specify the maximum number of yearly statistical records to retain.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Backup Tape Unit Name</td>
<td>1 to 8 characters</td>
<td>None</td>
<td>If you want to write repository backups to tape, specify the esoteric unit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>name for a tape device that is defined in your operating system environment.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To use the inherited value, clear the current value.</td>
</tr>
<tr>
<td>Days to Retain Tape</td>
<td>000 to 999</td>
<td>None</td>
<td>Specify the number of days to retain a repository backup that was written to</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>tape.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To use the inherited value, clear the current value.</td>
</tr>
</tbody>
</table>

Fast Path category in the Repository Management category
### Recovery & Change Accum parameters

The following table lists parameters in the Recovery & Change Accum category.

The Generate JCL function uses these parameters during generation of job steps that perform change accumulation or database recovery.
Table 28: Recovery and change accumulation parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAP Global Options</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the global options module that contains the option values to use during execution of the BMC Change Accumulation utility. In generated JCL for the utility, this name is used as the value of the PARMBLK keyword.</td>
</tr>
<tr>
<td>CAP CA Model</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the predefined dynamic allocation model to use for allocating output change accumulation data sets during execution of the BMC Change Accumulation utility.</td>
</tr>
<tr>
<td>RVP Global Options</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the global options module that contains the option values to use during execution of the BMC Recovery utility. In generated JCL for the utility, this name is used as the value of the PARMBLK keyword.</td>
</tr>
<tr>
<td>Use RVP Automatic Restart</td>
<td>Yes/No/Inherit</td>
<td>No</td>
<td>Specify whether created recovery JCL should contain statements to enable the Automatic Restart feature.</td>
</tr>
<tr>
<td>Checkpoint data set model</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the predefined model to use to create a CHKPT control statement. This model name is required if you enable the Automatic Restart feature.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CA Extract model</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the predefined model to use to create an XCUM control statement. This model name is optional if you enable the Automatic Restart feature.</td>
</tr>
<tr>
<td>Log Extract model</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the predefined model to use to create an XLOG control statement. This model name is optional if you enable the Automatic Restart feature.</td>
</tr>
<tr>
<td>Number of RVP IC Copies</td>
<td>Single, Dual, Inherit</td>
<td>Single</td>
<td>Specify whether to create one or two copies of the image copy data set during recovery.</td>
</tr>
<tr>
<td>RVP IC Copy 1 or Copy 2 Model</td>
<td>8 alphanumeric characters</td>
<td>None</td>
<td>Specify the name of the predefined dynamic allocation model to use for allocating output image copy data sets during execution of the BMC Recovery utility.</td>
</tr>
<tr>
<td>Rebuild Indexes category in the Recovery &amp; Change Accum category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Primary and Secondary Indexes</td>
<td>Rebuild all primary and secondary indexes, Rebuild primary and secondary indexes except HALDB secondary indexes, Do not rebuild primary and secondary indexes, Inherit</td>
<td>Rebuild all primary and secondary indexes</td>
<td>Specify whether created recovery JCL should contain keywords to enable rebuilding indexes during recovery.</td>
</tr>
<tr>
<td>Indirect List Data Sets (ILDS)</td>
<td>Rebuild the ILDS, Rebuild the ILDS only when needed, Do not rebuild the ILDS, Inherit</td>
<td>Rebuild the ILDS</td>
<td>Specify whether created recovery JCL should contain a keyword to enable rebuilding HALDB indirect list data sets (ILDSs) during recovery.</td>
</tr>
</tbody>
</table>
Customizing schedules

Schedules automate many processes for the BMC database administration products for IMS.

Overview of schedules

Schedules control the execution of functions that automate Database Advisor collection processing, Recovery Advisor collection and analysis processing, the E-mail Consolidation process, and repository management processes.

Indirectly, schedules control the frequency of Database Advisor analysis because data collection always results in analysis.

To conserve system resources, all Database Advisor scheduled functions include a start time and an end time. If a function is not complete by the end time, Database Advisor stops function processing. The next time that the function starts, processing resumes where it stopped.

Schedules are populated with suggested defaults, but you can customize the schedules to meet the requirements of each IMSplex in your environment.

Database Advisor schedules

The following Database Advisor functions can be scheduled at the IMSplex level:

- When one of the following functions is executed, Database Advisor collects data about your database, based on the data collection interval for the databases:
  - FF COLLECT (for full-function databases and HALDBs)
  - FF COLLECT HOURLY (for full-function databases and HALDBs)
  - FP COLLECT function (for Fast Path databases)

- When the FP SPACE COLLECT function is executed, Database Advisor starts collecting IOVF and SDEP space information for Fast Path databases.

- When the ENDOFDAY function is executed, Database Advisor analyzes your environment to determine whether any events of the day require action (such as resetting base values or increasing repository space allocation).

- When the DBSYNCH function is executed, Database Advisor detects changes between information in your RECONs with information in the repositories and
causes the registration process to update the repositories with current information.

- When the AUDIT function is executed, Database Advisor verifies that analysis has completed for all databases that have a defined collect schedule. You can schedule this function at the IMSPLEX level, or you can run it as a batch job. For more information, see “Auditing databases to ensure analysis” on page 170.

- The REXX EXEC SCHEDULE function controls when REXX EXEC batch jobs are generated for execution. When an exception is detected, you can queue a REXX EXEC to execute at a scheduled day and time.

For more information about the Collect process, see “Collect process” on page 140.

**Recovery Collect schedule**

You can schedule the RECOVERY COLLECT function at the IMSPLEX level.

When the RECOVERY COLLECT function is executed, Recovery Advisor reads and analyzes RECON data sets for the selected IMSPLEX. The analysis detects exceptions that are related to database operations, recoveries, and RECON availability and performance.

If the Exclude all objects threshold is set to Yes, Recovery Advisor detects RECON availability and performance exceptions only.

**E-mail Consolidation schedule**

You can use the E-MAIL CONSOLIDATION schedule to control the production of consolidated e-mail messages (which report the detection of exceptions).

**Repository management schedules**

Repository management schedules control the frequency of data management processes.

The following functions can be scheduled:

- The REP BACKUP function starts a process that backs up all repositories. You can schedule this function at the IMSPLEX level.
The PURGE function starts an internal process to purge (delete) data from the repositories, based on user-defined repository management parameters. You can schedule this function at the host level.

For more information about repository management, see the Database Products for IMS Customization Guide.

## Summary of scheduled functions

The following table summarizes functions that you can schedule through the console.

### Table 29: Scheduled function summary

<table>
<thead>
<tr>
<th>Function</th>
<th>Active by default?</th>
<th>Default schedule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUDIT</td>
<td>No</td>
<td>None</td>
<td>The AUDIT function verifies that analysis has completed for all databases that have a defined collection schedule. For more information, see “Auditing databases to ensure analysis” on page 170.</td>
</tr>
<tr>
<td>DBSYNCH</td>
<td>No</td>
<td>None</td>
<td>The Database Advisor DBSYNCH function synchronizes registry information in Database Advisor with current information in the RECONs, detects any changes, and causes the registration process to update the repositories with current information. The difference between the DBSYNCH function and the Register object menu command is that the DBSYNCH function adds, changes, and deletes information from the repositories so that the repositories reflect all (and only) current information. The Register object menu command only adds information to the repositories.</td>
</tr>
<tr>
<td>E-MAIL CONSOLIDATION</td>
<td>No</td>
<td>None</td>
<td>The E-MAIL CONSOLIDATION function controls whether and when consolidated e-mail messages, which provide information about detected exceptions, are sent to designated recipients on a scheduled basis. This function is related to the E-Mail/WTO parameters that are described in “E-mail / WTO / REXX EXEC parameters” on page 253. You can edit this function at the IMSPLEX level.</td>
</tr>
<tr>
<td>Function</td>
<td>Active by default?</td>
<td>Default schedule</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------</td>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| ENDOFDAY        | Yes                | Daily, 1:00 A.M to 4:00 A.M. | The Database Advisor ENDOFDAY function collects historical information for full-function databases that were reloaded during the day, updates base values, and initiates analysis of databases that have updated values.

A database is included in the ENDOFDAY function if any of the following events occur without a pointer checker being run afterwards:

- The database load utility is executed.
- An IDCAMS REPRO is performed on a database data set.
- A database is recovered. |
| FF COLLECT      | Yes                | Daily, 1:00 A.M to 4:00 A.M. | The Database Advisor FF COLLECT function collects statistics about full-function databases and HALDBs and writes statistics detail records. This function is scheduled automatically by default.

A database is included in the FF COLLECT function if the time interval between the current date and the date of the last data collection is greater than the value of the Data Collection Interval parameter.

Base values are set in this function if the database had no existing base values. Database Advisor uses base values to assess changes in an object by comparing base values with current state information. By monitoring these changes, Database Advisor identifies threshold violations.

The Data Collection Interval parameter is used with the FF COLLECT schedule (and the FP COLLECT schedule). The default value is 28 days.

A job to execute the FF COLLECT function processes a maximum of 240 databases. If more than 240 require processing, Advisor initiates multiple FF COLLECT jobs, as needed. To ensure that these jobs have unique names and can run concurrently, you must add the GROUP SCHDTASK(M) keyword to your CPCxINI procedure. For more information, see the Database Products for IMS Customization Guide.
<table>
<thead>
<tr>
<th>Function</th>
<th>Active by default?</th>
<th>Default schedule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FF COLLECT HOURLY</td>
<td>No</td>
<td>None</td>
<td>The Database Advisor FF COLLECT HOURLY function collects statistics about full-function databases and HALDBs and writes statistics detail records. This function is not scheduled automatically by default. Base values are set in this function if the database had no existing base values. Database Advisor uses base values to assess changes in an object by comparing base values with current state information. By monitoring these changes, Database Advisor identifies threshold violations. The Activate FF Hourly Collect parameter is used with the FF COLLECT HOURLY schedule. No default value is provided.</td>
</tr>
<tr>
<td>FP COLLECT</td>
<td>No</td>
<td>Daily, 1:00 A.M to 4:00 A.M.</td>
<td>The Database Advisor FP COLLECT function collects statistics about Fast Path databases and writes statistics detail records. This function is not scheduled automatically by default; however, other schedule parameters are the same as for the FF COLLECT function. Before you schedule the FP COLLECT function, see the installation requirements in the Installation System documentation. A Fast Path database is included in the FP COLLECT function if the time interval between the current date and the date of the last data collection is greater than the Data Collection Interval parameter. The Data Collection Interval parameter is used with the FP COLLECT schedule (and the FF COLLECT schedule). The default value is 28 days. A job to execute the FP COLLECT function processes a maximum of 240 areas. If more than 240 require processing, Advisor initiates multiple FP COLLECT jobs, as needed. To ensure that these jobs have unique names and can run concurrently, you must add the GROUP SCHDTASK(M) keyword to your CPCxINI procedure. For more information, see the Database Products for IMS Customization Guide.</td>
</tr>
</tbody>
</table>
The Database Advisor FP SPACE COLLECT function collects space-related statistics for Fast Path databases and writes space detail records. This function is not scheduled automatically by default. Before you schedule the FP SPACE COLLECT function, see the installation requirements in the Installation System documentation.

The FP SPACE COLLECT function collects IOVF and SDEP space usage statistics, which are a subset of the information that is collected by the FP COLLECT function. By collecting this subset instead of the full range of statistics, Fast Path areas can be processed quickly.

A Fast Path database is included in the FP SPACE COLLECT function if the difference between the current time and the last time that data was collected is greater than the Fast Path Space Data Collection Interval parameter.

**WARNING:** Because of the potential for high resource consumption when the FP SPACE COLLECT function is executed, you should not specify Yes for this option at the RECON level or a higher level. You should specify Yes for this option only at the database group level or the area level.

A Fast Path database is included in the FP SPACE COLLECT function if the Activate Fast Path Space Data Collection parameter is set to Yes.
<table>
<thead>
<tr>
<th>Function</th>
<th>Active by default?</th>
<th>Default schedule</th>
<th>Description</th>
</tr>
</thead>
</table>
| REP BACKUP            | Yes                | Daily, midnight to 1:00 A.M. | The repository REP BACKUP function backs up repository data (statistical information, analysis options, and processing information).  
This function analyzes repository data groups to determine whether the maximum extents threshold has been violated.  
If the number of extents in a data group meets or exceeds the threshold, the following tasks are performed:  
■ The function that is associated with the data group is stopped.  
■ The data group is rebuilt with an increased space allocation of 20 percent.  
■ The function that is associated with the data group is restarted.  
The maximum extents threshold (used with the REP BACKUP function) is used exclusively as a repository management threshold. The maximum extents threshold is not related to the MAXM OSAM/VSAM Extents threshold that is defined in the console for user databases. |
| REXX EXEC SCHEDULE    | No                 | None                          | The REXX EXEC SCHEDULE function controls when REXX EXEC batch jobs are generated for execution. When an exception is detected, you can queue a REXX EXEC to execute at a scheduled day and time. |

**Editing schedules**

You can control the frequency of each scheduled function by editing the corresponding schedule.

**Note**

When you have implemented a new schedule or edited an existing schedule, remember to verify that scheduled tasks have been executing as expected.

**To edit schedules**

1. Open the schedule for the function that you want to edit:
To edit the schedule for backing up repositories, right-click a host object in the Navigation window and choose **Configuration => Schedule Repository Backups**.

To edit the schedule for all other functions, right-click an IMSPLEX in the Navigation window and choose **Configuration => Schedules**.

The Schedules window (Figure 45 on page 286) is displayed. For a description of the functions that are listed in this window, see “Summary of scheduled functions” on page 281.

**Figure 45: Schedules window**

2 In the Schedules window, double-click the row of the function that you want to schedule.
The Edit Schedule dialog box (Figure 46 on page 287) is displayed. Figure 46 on page 287 shows the dialog box for the PURGE function, but all functions use the same parameters.

Figure 46: Edit Schedule dialog box

3 In the Edit Schedule dialog box, edit data collection information and click Save.

For more information about each parameter, including descriptions, valid values, and suggested defaults, see the online Help.

Checking scheduled function completion codes

When a scheduled function executes, you can determine whether the function completed successfully by looking at its return code (completion code).

The completion code is listed in the Edit Schedule dialog box for each function.

To view the completion code

1 In the Navigation window, right-click the IMSPLEX for which you want to edit schedules and choose Configuration => Schedules.

The Schedules window (Figure 44 on page 230) is displayed.
2. In the Schedules window, double-click the row of the function for which you want to view the completion code.

The Edit Schedule dialog box is displayed, and the completion code is listed in the Last Completion Code field.

Table 30 on page 288 lists the completion (return) codes for the following functions:

- E-MAIL CONSOLIDATION
- FF COLLECT
- FF COLLECT HOURLY
- FP COLLECT
- FP SPACE COLLECT
- ENDOFDAY
- PURGE
- REP BACKUP
- DBSYNCH
- REXX EXEC SCHEDULE

<table>
<thead>
<tr>
<th>Return code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The function completed successfully.</td>
</tr>
<tr>
<td>4</td>
<td>The function completed successfully, but no databases were found to process. For the second step of COLLECT or ENDOFDAY functions, return code RC=4 might indicate that a data set was missing.</td>
</tr>
<tr>
<td>n &gt; 4</td>
<td>An error occurred. View the CPCLOG and, if necessary, the CPCTRACE for the Database Advisor address space and the batch task.</td>
</tr>
</tbody>
</table>

Table 31 on page 288 lists the completion (return) codes for the RECOVERY COLLECT function.

<table>
<thead>
<tr>
<th>Return code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>No errors or warnings occurred.</td>
</tr>
<tr>
<td>14</td>
<td>The function detected an error condition that was related to the RECON data sets (for example, a RECON data set has been discarded). If possible, the function finished executing.</td>
</tr>
<tr>
<td>16</td>
<td>An error occurred. Function execution was unable to continue. Contact BMC Customer Support for assistance.</td>
</tr>
</tbody>
</table>
Using program CPCTSCHD to initiate Advisor scheduled tasks

An alternative technique for initiating Advisor scheduled tasks is to use a batch job to execute program CPCTSCHD.

Executing CPCTSCHD initiates the specified task in the same way that the Advisor scheduling system would initiate it, except the initiation is immediate. You can use CPCSCHD to initiate any task instead of or in addition to initiating that task through the Advisor scheduling system. Executing CPCTSCHD does not affect established schedules in the Advisor scheduling system.

JCL and control statements for program CPCTSCHD

Sample JCL to execute program CPCTSCHD is located in member CPCSCHD in the sample library.

The CPCSYSIN control statement data set contains the SCHEDULE command and required keywords. A value, enclosed in parentheses, follows each keyword.

To specify control statement keywords for CPCTSCHD

1. For the SCHEDULE command, specify the required TASKNAME(taskName) keyword with one of the following values for taskName:

<table>
<thead>
<tr>
<th>TASKNAME value</th>
<th>Description</th>
<th>Required keywords for this value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUDIT</td>
<td>Verify that analysis has completed for all databases that have a defined collection schedule. For more information, see “Auditing databases to ensure analysis” on page 170.</td>
<td>FROMRECON</td>
</tr>
<tr>
<td>BACKUP</td>
<td>Back up repository data.</td>
<td>None</td>
</tr>
<tr>
<td>DBSYNC</td>
<td>Synchronize registry information in Database Advisor with current information in the RECONs, detect any changes, and cause the registration process to update the repositories with current information.</td>
<td>FROMRECON</td>
</tr>
<tr>
<td>TASKNAME value</td>
<td>Description</td>
<td>Required keywords for this value</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>EMAILCON</td>
<td>Send consolidated e-mail messages, which provide information about detected exceptions, to designated recipients. This task uses the parameter values that are described in “E-mail / WTO / REXX EXEC parameters” on page 253.</td>
<td>FROMRECON</td>
</tr>
</tbody>
</table>
| EODCOLLECT     | Collect historical information for full-function databases that were reloaded during the day, update base values, and initiate analysis of databases that have updated values. A database is included in this task if any of the following events occur without a pointer checker being run afterwards:  
  ■ The database load utility is executed.  
  ■ An IDCAMS REPRO is performed on a database data set.  
  ■ A database is recovered. | FROMRECON                        |
<p>| FFCOLLECT      | Collect statistics about full-function databases and HALDBs and write statistics detail records. A database is included in this task if the time interval between the current date and the date of the last data collection is greater than the value of the Data Collection Interval parameter. | FROMRECON                        |</p>
<table>
<thead>
<tr>
<th>TASKNAME value</th>
<th>Description</th>
<th>Required keywords for this value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPCOLLECT</td>
<td>Collect statistics about Fast Path databases and write statistics detail records. A database is included in this task if the time interval between the current date and the date of the last data collection is greater than the value of the Data Collection Interval parameter.</td>
<td>FROMRECON</td>
</tr>
</tbody>
</table>
| FSPACE         | Collect space-related statistics for Fast Path databases and write space detail records. This function is not scheduled automatically by default. Before you schedule this task, see the installation requirements in the the Installation System documentation. A Fast Path database is included in this task if the following conditions exist:  
  - The difference between the current time and the last time data was collected is greater than the Fast Path Space Data Collection Interval parameter.  
  - The Activate Fast Path Space Data Collection parameter is set to Yes. | FROMRECON |
| HCOLLECT       | Collect statistics about full-function databases and HALDBs and write statistics detail records. A database is included in this task if the Activate FF Hourly Collect parameter is set to Yes. | FROMRECON |
| ICOLLECT       | Immediately collect and analyze data for full-function and HALDB databases. | FROMRECON, DBDNAME, PARTNAME |
### TASKNAME value

<table>
<thead>
<tr>
<th>TASKNAME value</th>
<th>Description</th>
<th>Required keywords for this value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFPCOLLECT</td>
<td>Immediately collect statistics about Fast Path databases and write statistics detail records. Before you schedule this task, see the installation requirements in the Installation System documentation.</td>
<td>FROMRECON, DBDNAME, PARTNAME</td>
</tr>
<tr>
<td>PURGE</td>
<td>Start an internal process to purge (delete) repository data, based on site values that specify how long to retain statistical information.</td>
<td>FROMRECON</td>
</tr>
<tr>
<td>RCOLLECT</td>
<td>Execute Recovery Advisor collection and analysis processing. This process scans the RECON for conditions that can affect the recoverability of IMS data and the performance of RECON data sets.</td>
<td>FROMRECON</td>
</tr>
</tbody>
</table>

1. If required for the specified task, specify the **DBDNAME(name)** keyword to provide the DBD name of the database to process.

2. If required for the specified task, specify the **PARTNAME(name)** keyword to provide the name of the partition or area to process.

3. If required for the specified task, specify the **FROMRECON(reconDataSetName)** keyword to provide the data set name of the RECON that defines the objects to process.

### Security for program CPCTSCHD

To execute a scheduled task through program CPCTSCHD, you must have access to the resource class FACILITY.

You can change the default entity name in the CPCOPT member if the name does not conform to your security standards. The default entity name is SCHEDULE.TASKS."reconDataSetName".

To allow access to all RECONs, you can specify an asterisk (*) for the RECON data set name. To schedule the backup (because it encompasses all RECONs), the entity name is SCHEDULE.TASKS.SYSPLEX.

Users must be granted READ access to pass the security check.
Configuring dynamic allocation models and matrixes

Use the following procedure to configure dynamic allocation models and matrixes. They provide a simple, flexible method for managing output data sets (such as image copy, change accumulation, extract, and checkpoint data sets) for backup, change accumulation, and recovery processes. Through automation and a central point of definition, models and matrixes produce superior results while reducing the burden of specifying and maintaining JCL for the utilities.

Dynamic allocation models contain values that define how to allocate output data sets. For an individual output data set, models provide complete control over the following items:

- Data set name
- Device type
- Space usage
- Retention period or expiration date
- Other allocation parameters

Models support Storage Management Subsystem (SMS) techniques for allocating data sets.

For the BMC Image Copy utility, you can define a model matrix, based on the size of the input data set to be copied. A matrix defines a series of paired values. Each pair consists of a data set size and the name of the dynamic allocation model to use for the output data set if the input data set is smaller than the defined size.

**Note**

To work with dynamic allocation models and matrixes in the console, you must have a license for the Backup and Recovery Solution for IMS product or the MAXM Database Advisor for IMS product.

**To configure dynamic allocation models and matrixes**

1. Right-click the IMSPLEX for which you want to manage dynamic allocation models and matrixes and select Configuration => Backup and Recovery Models.

   For more information, see the online Help topics related to dynamic allocation models and matrixes.

2. To use a model or matrix for an output data set, specify the model or matrix name as the value of the ICP Image Copy Model parameter (in the Maintain Parameters window) or as the value of a keyword on an applicable control statement.
For more information, see the utility user guide.
Handling exceptions

This section contains information about working with Advisor exceptions and the Exception List.

Overview of exceptions

Advisor reports detected problems as exceptions.

Exception records

Advisor analysis processes store information about exceptions as exception records in the repository:

- Database Advisor produces an exception record for each threshold that is evaluated. The record identifies the database that was analyzed, the current status of the threshold, and the dates on which the threshold is expected to reach each status severity. When the next analysis occurs, the status of the exception is updated. The date on which the exception is forecast to reach the next status level (for example, from warning to critical) is also projected.
  
  Determination of the historical base point is a significant factor in Fast Path exception processing. For more information, see “Fast Path DEDB base values” on page 144.

- Recovery Advisor produces an exception record for each recovery threshold problem that is found for databases, change accumulation data sets, and log data sets. Because Recovery Advisor does not keep historical information, recovery threshold exceptions are cleared at the beginning of each execution of the process.

- Recovery Advisor produces an exception record for each RECON threshold that is evaluated. Recovery Advisor evaluates the RECON threshold, current status, and growth rate to determine the dates on which the current value is predicted to reach or exceed the threshold value and assigns a severity level in accordance with this date.
Exception status types

Exceptions can have the following degrees of severity, as indicated by the status type.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Status type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>!</td>
<td>Warning (yellow)</td>
<td>An exception has been detected but might not need to be corrected immediately, or an exception is predicted but has not yet occurred.</td>
</tr>
<tr>
<td>✗</td>
<td>Critical (red)</td>
<td>A threshold has been met or exceeded, and the situation requires corrective action.</td>
</tr>
<tr>
<td>☠</td>
<td>Dead (black)</td>
<td>An operating system limit has been reached, and the situation requires immediate action.</td>
</tr>
<tr>
<td>✅</td>
<td>Not analyzed (orange)</td>
<td>The AUDIT function found a potential problem related to the data collection or analysis of the object.</td>
</tr>
</tbody>
</table>

When an exception is forecast for a database, the icon for the database object in the Navigation window changes to include the exception severity. Table 32 on page 296 lists the severity types that are used for database objects when an exception is reported.

Table 32: Exception severity types for databases

<table>
<thead>
<tr>
<th>Icon</th>
<th>Severity type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>📆</td>
<td>Database with warning exception</td>
<td>Warning exceptions (and no critical or dead exceptions) were found for the database.</td>
</tr>
<tr>
<td>📆</td>
<td>Database with critical exception</td>
<td>Critical exceptions were found for the database.</td>
</tr>
<tr>
<td>📆</td>
<td>Database with dead exception</td>
<td>Dead exceptions were found for the database.</td>
</tr>
</tbody>
</table>

Exception types

Advisor can detect the types of exceptions listed in the following table.

The table lists the name of the exception, the type of object for which the exception can be detected, the component that must be licensed for the exception to be detected, and reference that provides more details about the exception.
<table>
<thead>
<tr>
<th>Exception</th>
<th>Object type</th>
<th>Licensed component</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREA DOVF UTILIZATION TOO HIGH</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA DOVF UTILIZATION TOO HIGH exception” on page 150</td>
</tr>
<tr>
<td>AREA FRAGMENTATION TOO HIGH</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA FRAGMENTATION TOO HIGH exception” on page 150</td>
</tr>
<tr>
<td>AREA I/O PROBABILITY TOO HIGH</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA I/O PROBABILITY TOO HIGH exception” on page 151</td>
</tr>
<tr>
<td>AREA IOVF SPACE LOW</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA IOVF SPACE LOW exception” on page 151</td>
</tr>
<tr>
<td>AREA RAA UTILIZATION TOO HIGH</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA RAA UTILIZATION TOO HIGH exception” on page 151</td>
</tr>
<tr>
<td>AREA SDEP FREE SPACE TOO LOW</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>“AREA SDEP FREE SPACE TOO LOW exception” on page 151</td>
</tr>
<tr>
<td>AREA USED PERCENT EXCEEDED</td>
<td>Fast Path area</td>
<td>Database Advisor</td>
<td>AREA USED PERCENT EXCEEDED exception on page 152</td>
</tr>
<tr>
<td>BACKOUT NEEDED</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“BACKOUT NEEDED exception” on page 178</td>
</tr>
<tr>
<td>CA DATA SET NOT CATALOGED</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“CA DATA SET NOT CATALOGED exception” on page 179</td>
</tr>
<tr>
<td>CA RUN IN ERROR</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“CA RUN IN ERROR exception” on page 179</td>
</tr>
<tr>
<td>DBDS IN MULTIPLE DBDS GROUPS</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“DBDS IN MULTIPLE DBDS GROUPS exception” on page 179</td>
</tr>
<tr>
<td>DBDS NOT IN ANY DBDS GROUP</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“DBDS NOT IN ANY DBDS GROUP exception” on page 180</td>
</tr>
</tbody>
</table>
### Exception Overview

<table>
<thead>
<tr>
<th>Exception</th>
<th>Object type</th>
<th>Licensed component</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBDS NOT IN CHANGE ACCUM GROUP</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“DBDS NOT IN CHANGE ACCUM GROUP exception” on page 180</td>
</tr>
<tr>
<td>DSN SIZE EXCEEDS THRESHOLD</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“DSN SIZE EXCEEDS THRESHOLD exception” on page 203</td>
</tr>
<tr>
<td>EEQE(S) FOUND</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“EEQE(S) FOUND exception” on page 180</td>
</tr>
<tr>
<td>FSE ERROR</td>
<td>Database</td>
<td>Database Advisor</td>
<td>FSE ERROR exception on page 152</td>
</tr>
<tr>
<td>GENMAX BELOW THRESHOLD</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“GENMAX BELOW THRESHOLD exception” on page 181</td>
</tr>
<tr>
<td>GENMAX EXCEEDED THRESHOLD</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“GENMAX EXCEEDED THRESHOLD exception” on page 181</td>
</tr>
<tr>
<td>GRPMAX BELOW THRESHOLD</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“GRPMAX BELOW THRESHOLD exception” on page 181</td>
</tr>
<tr>
<td>GRPMAX EXCEEDED THRESHOLD</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“GRPMAX EXCEEDED THRESHOLD exception” on page 182</td>
</tr>
<tr>
<td>HALDB PARTITION NOT INITIALIZED</td>
<td>HALDB</td>
<td>Recovery Advisor</td>
<td>“HALDB PARTITION NOT INITIALIZED exception” on page 182</td>
</tr>
<tr>
<td>HIDAM PRIMARY INDEX ERROR</td>
<td>Database</td>
<td>Database Advisor</td>
<td>HIDAM PRIMARY INDEX ERROR exception on page 152</td>
</tr>
<tr>
<td>IC DATA SET IN ERROR</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“IC DATA SET IN ERROR exception” on page 182</td>
</tr>
<tr>
<td>IC DATA SET NOT CATALOGED</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“IC DATA SET NOT CATALOGED exception” on page 183</td>
</tr>
<tr>
<td>IC RECOMMENDED BY CA</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“IC RECOMMENDED BY CHANGE ACCUM exception” on page 183</td>
</tr>
<tr>
<td>IMAGE COPY NEEDED</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“IMAGE COPY NEEDED exception” on page 184</td>
</tr>
<tr>
<td>Exception</td>
<td>Object type</td>
<td>Licensed component</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>---------------------------</td>
<td>--------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IMAGE COPY RECOMMENDED</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“IMAGE COPY RECOMMENDED exception” on page 184</td>
</tr>
<tr>
<td>INCORRECT CI SIZE exception</td>
<td>VSAM data set</td>
<td>Database Advisor</td>
<td>INCORRECT CI SIZE exception on page 153</td>
</tr>
<tr>
<td>INSUFF ALLOC FREE SPACE (LS)</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFF ALLOC FREE SPACE (LS) exception” on page 153</td>
</tr>
<tr>
<td>INSUFF TOTAL FREE SPACE (LS)</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFF TOTAL FREE SPACE (LS) exception” on page 153</td>
</tr>
<tr>
<td>INSUFFICIENT ALLOCATED FREE SPACE</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFFICIENT ALLOCATED FREE SPACE exception” on page 153</td>
</tr>
<tr>
<td>INSUFFICIENT FREE BLOCKS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFFICIENT FREE BLOCKS exception” on page 154</td>
</tr>
<tr>
<td>INSUFFICIENT FREE SPACE</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“INSUFFICIENT FREE SPACE exception” on page 203</td>
</tr>
<tr>
<td>INSUFFICIENT FREE SPACE IN RAA</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFFICIENT FREE SPACE IN RAA exception” on page 154</td>
</tr>
<tr>
<td>INSUFFICIENT SPACE</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFFICIENT SPACE exception” on page 155</td>
</tr>
<tr>
<td>INSUFFICIENT TOTAL FREE SPACE</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“INSUFFICIENT TOTAL FREE SPACE exception” on page 155</td>
</tr>
<tr>
<td>I/O PROBABILITY TOO HIGH</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“I/O PROBABILITY TOO HIGH exception” on page 155</td>
</tr>
<tr>
<td>MAXIMUM OSAM ALLOC EXCEEDED</td>
<td>OSAM data set</td>
<td>Database Advisor</td>
<td>MAXIMUM OSAM ALLOC EXCEEDED exception on page 156</td>
</tr>
<tr>
<td>MAXIMUM VSAM ALLOC EXCEEDED</td>
<td>VSAM data set</td>
<td>Database Advisor</td>
<td>MAXIMUM VSAM ALLOC EXCEEDED exception on page 156</td>
</tr>
<tr>
<td>MIN # OF CA RUNS NOT AVAILABLE</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“MIN # OF CA RUNS NOT AVAILABLE exception” on page 185</td>
</tr>
<tr>
<td>Exception</td>
<td>Object type</td>
<td>Licensed component</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------</td>
<td>--------------------</td>
<td>----------------------------------------------------------------</td>
</tr>
<tr>
<td>MIN # OF IC’S NOT AVAILABLE</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“MIN # OF IC’S NOT AVAILABLE exception” on page 185</td>
</tr>
<tr>
<td>MORE LOGS THAN THRESHOLD</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“MORE LOGS THAN THRESHOLD exception” on page 186</td>
</tr>
<tr>
<td>NO AREA DATA SET AVAILABLE</td>
<td>Fast Path area</td>
<td>Recovery Advisor</td>
<td>“NO AREA DATA SET AVAILABLE exception” on page 186</td>
</tr>
<tr>
<td>NO AREA DATA SET DEFINED</td>
<td>Fast Path area</td>
<td>Recovery Advisor</td>
<td>“NO AREA DATA SET DEFINED exception” on page 186</td>
</tr>
<tr>
<td>NO CA RUN WITHIN TIME RANGE</td>
<td>Change accumulation group</td>
<td>Recovery Advisor</td>
<td>“NO CA RUN WITHIN TIME RANGE exception” on page 186</td>
</tr>
<tr>
<td>NO IC WITHIN TIME RANGE</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“NO IC WITHIN TIME RANGE exception” on page 187</td>
</tr>
<tr>
<td>NO SECIC WITHIN TIME RANGE</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“NO SECIC WITHIN TIME RANGE exception” on page 188</td>
</tr>
<tr>
<td>NO SECLOG WITHIN TIME RANGE</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“NO SECLOG WITHIN TIME RANGE exception” on page 188</td>
</tr>
<tr>
<td>NO SECSLD WITHIN TIME RANGE</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“NO SECSLD WITHIN TIME RANGE exception” on page 189</td>
</tr>
<tr>
<td>NO VALID IMAGE COPY</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“NO VALID IMAGE COPY exception” on page 189</td>
</tr>
<tr>
<td>POINTER ERROR</td>
<td>Database</td>
<td>Database Advisor</td>
<td>POINTER ERROR exception on page 156</td>
</tr>
<tr>
<td>PRILOG ENTRY IN ERROR</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“PRILOG ENTRY IN ERROR exception” on page 190</td>
</tr>
<tr>
<td>PRILOG ENTRY NOT CATALOGED</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“PRILOG ENTRY NOT CATALOGED exception” on page 191</td>
</tr>
<tr>
<td>PRILOG SIZE EXCEEDS THRESHOLD</td>
<td>Log</td>
<td>Recovery Advisor or Database Advisor</td>
<td>PRILOG SIZE EXCEEDS THRESHOLD exception on page 157</td>
</tr>
</tbody>
</table>

Overview of exceptions
<table>
<thead>
<tr>
<th>Exception</th>
<th>Object type</th>
<th>Licensed component</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRISLD ENTRY IN ERROR</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“PRISLD ENTRY IN ERROR exception” on page 190</td>
</tr>
<tr>
<td>PRISLD ENTRY NOT CATALOGED</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“PRISLD ENTRY NOT CATALOGED exception” on page 191</td>
</tr>
<tr>
<td>PROHIBIT AUTHORIZATION FOUND</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“PROHIBIT AUTHORIZATION FOUND exception” on page 191</td>
</tr>
<tr>
<td>RAA TOTAL FREE SPACE TOO SMALL</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“RAA TOTAL FREE SPACE TOO SMALL exception” on page 157</td>
</tr>
<tr>
<td>READ ONLY</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“READ ONLY exception” on page 191</td>
</tr>
<tr>
<td>RECLAIMABLE FREESPACE EXCEEDED</td>
<td>Index database</td>
<td>Database Advisor</td>
<td>RECLAIMABLE FREESPACE EXCEEDED exception on page 157</td>
</tr>
<tr>
<td>RECLAIMABLE IOVF EXCEEDED</td>
<td>Fast path database</td>
<td>Database Advisor</td>
<td>RECLAIMABLE IOVF EXCEEDED exception on page 157</td>
</tr>
<tr>
<td>RECLAIMABLE SPACE EXCEEDED</td>
<td>Index database</td>
<td>Database Advisor</td>
<td>RECLAIMABLE SPACE EXCEEDED exception on page 158</td>
</tr>
<tr>
<td>RECON DISCARDED / UNAVAILABLE</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“RECON DISCARDED / UNAVAILABLE exception” on page 204</td>
</tr>
<tr>
<td>RECOVERY NEEDED</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“RECOVERY NEEDED exception” on page 192</td>
</tr>
<tr>
<td>RECOVERY NEEDED COUNT INVALID</td>
<td>DBD, DEDB, or HALDB partition</td>
<td>Recovery Advisor</td>
<td>RECOVERY NEEDED COUNT INVALID exception on page 158</td>
</tr>
<tr>
<td>RECOVPD BELOW THRESHOLD</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“RECOVPD BELOW THRESHOLD exception” on page 192</td>
</tr>
<tr>
<td>RECOVPD EXCEEDED THRESHOLD</td>
<td>Database</td>
<td>Recovery Advisor</td>
<td>“RECOVPD EXCEEDED THRESHOLD exception” on page 193</td>
</tr>
<tr>
<td>SECLOG ENTRY IN ERROR</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“SECLOG ENTRY IN ERROR exception” on page 193</td>
</tr>
<tr>
<td>Exception</td>
<td>Object type</td>
<td>Licensed component</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>----------------------------</td>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SECLOG ENTRY NOT CATALOGED</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“SECLOG ENTRY NOT CATALOGED exception” on page 193</td>
</tr>
<tr>
<td>SECSLD ENTRY IN ERROR</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“SECLOG ENTRY IN ERROR exception” on page 193</td>
</tr>
<tr>
<td>SECSLD ENTRY NOT CATALOGED</td>
<td>Log</td>
<td>Recovery Advisor</td>
<td>“SECLOG ENTRY NOT CATALOGED exception” on page 193</td>
</tr>
<tr>
<td>TOO FEW DOVF CONTROL INTERVALS</td>
<td>Fast Path database</td>
<td>Database Advisor</td>
<td>TOO FEW DOVF CONTROL INTERVALS exception on page 158</td>
</tr>
<tr>
<td>TOO FEW EMPTY RAPS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>TOO FEW EMPTY RAPS exception on page 158</td>
</tr>
<tr>
<td>TOO FEW IOVF BLOCKS</td>
<td>Fast Path database</td>
<td>Database Advisor</td>
<td>TOO FEW IOVF BLOCKS exception on page 159</td>
</tr>
<tr>
<td>TOO MANY CA SPLITS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY CA SPLITS exception” on page 159</td>
</tr>
<tr>
<td>TOO MANY CA SPLITS</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“TOO MANY CA SPLITS exception” on page 204</td>
</tr>
<tr>
<td>TOO MANY CI SPLITS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY CI SPLITS exception” on page 159</td>
</tr>
<tr>
<td>TOO MANY CI SPLITS</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“TOO MANY CI SPLITS exception” on page 205</td>
</tr>
<tr>
<td>TOO MANY EXTENTS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY EXTENTS exception” on page 159</td>
</tr>
<tr>
<td>TOO MANY EXTENTS</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“TOO MANY EXTENTS exception” on page 205</td>
</tr>
<tr>
<td>TOO MANY FRAGMENTED DATABASE RECORDS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY FRAGMENTED DATABASE RECORDS exception” on page 160</td>
</tr>
<tr>
<td>TOO MANY ROOTS IN OVERFLOW</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY ROOTS IN OVERFLOW exception” on page 161</td>
</tr>
<tr>
<td>TOO MANY SEGMENT SPLITS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY SEGMENT SPLITS exception” on page 161</td>
</tr>
<tr>
<td>TOO MANY SPLITS ACROSS BLOCKS</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY SPLITS ACROSS BLOCKS exception” on page 162</td>
</tr>
<tr>
<td>Exception</td>
<td>Object type</td>
<td>Licensed component</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------</td>
<td>-------------------------------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>TOO MANY VOLUMES</td>
<td>Database</td>
<td>Database Advisor</td>
<td>“TOO MANY VOLUMES exception” on page 162</td>
</tr>
<tr>
<td>TOO MANY VOLUMES</td>
<td>RECON data set</td>
<td>Recovery Advisor or Database Advisor</td>
<td>“TOO MANY VOLUMES exception” on page 206</td>
</tr>
<tr>
<td>UNALLOCATED DATASET</td>
<td>Database</td>
<td>Database Advisor</td>
<td>UNALLOCATED DATASET exception on page 162</td>
</tr>
<tr>
<td>UNINITIALIZED DATASET</td>
<td>Database</td>
<td>Database Advisor</td>
<td>UNINITIALIZED DATASET exception on page 162</td>
</tr>
</tbody>
</table>

**Working with exceptions and the Exception List**

Objects (databases, change accumulations, logs, and RECONS) for which warning, critical, or dead exceptions were found are listed on the Exception List.

For Database Advisor and RECON exceptions, the Exception List also reports the dates on which an exception is projected to reach the next higher status severity, if applicable.

When an exception is reported to the Exception List, a current or potential problem might exist. You can start working with exceptions by reviewing individual exceptions for an object, identifying the status of an exception, and understanding how an exception reaches its status. With this knowledge, you can determine appropriate actions to correct or prevent a problem.
Use the Exception List window (Figure 47 on page 304) to manage exceptions.

**Figure 47: Exception List window**

When an exception is detected, you can queue a REXX EXEC to execute at a scheduled day and time. For more information, see the information about the REXX EXEC SCHEDULE function in the Customizing parameters and schedules on page 225 chapter.

---

**Opening the Exception List**

You can open the Exception List at various levels in the Navigation window.

The level at which you open the Exception List depends on the scope of the exceptions that you want to display in the Exception List:
If you open the Exception List from a host connection object, all exceptions that were found across the entire sysplex are displayed in the Exception List.

If you open the Exception List from an IMSPLEX object, all exceptions that were found across the IMSPLEX are displayed in the Exception List.

If you open the Exception List from a database group object, all exceptions that were found across that database group are displayed in the Exception List.

**To open the Exception List**

1. Right-click the object (sysplex, IMSPLEX, or database group) in the Navigation window and choose Exception List.

The Exception List window (Figure 47 on page 304) is displayed.

**Reading the Exception List**

The Exception List includes a Summary table (top) and a Detail table (bottom).

The Summary table lists all objects for which exceptions were reported. The Detail table lists the specific exceptions for the object that is selected in the Summary table. For database objects, exceptions can occur at the database level or the data set level.

Key information in the Summary table is the roll-up status in the Status column (which is not labeled). An icon in the Highest Status column indicates the status of the most severe exception that has been reported for that object.

The Detail table shows dates in the Warning Date column, the Critical Date column, and the Dead Date column. For Database Advisor and RECON exceptions, the dates in these columns indicate when a threshold is expected to reach the indicated status. For recovery exceptions, the date in the Warning Date column, the Critical Date column, or the Dead Date column indicates when the exception was detected. The specific exception type is identified in the Description column.

**Filtering the Exception List by exception status types**

The default display of the Exception List shows all exception status types (warning, critical, and dead) and all object types. You can limit the displayed exceptions by status (severity), object type, and exception category. Only the exceptions that match all of the filtering criteria are retrieved from the Advisor repositories and displayed in the Exception List window.
**Note**

If the **Show filter dialog initially** option is selected in this dialog, the console displays the dialog automatically when you open the Exception List window. The filter does not apply to the Solutions window, which you can access from the Exception List window.

---

**To filter the Exception List**

1. In the Exception List window, click **Filter**.

   The Filter for Exception List dialog box ([Figure 48 on page 306](#)) is displayed.

---

**Figure 48: Filter for exception list dialog box**
2 Select one of the following options for the **Lowest Status Filter** field:
- To view all exceptions, select **Warning**.
- To view exceptions with a critical or dead status, select **Critical**.
- To view exceptions with a dead status, select **Dead**.
- To view exceptions with a not-analyzed status, select **Not Analyzed**.

3 Select types of objects to include:
- To view exceptions for all objects, select **All Types**.
- To view exceptions for specific types of objects, select **Specific** and check all types that you want to view (full-function databases, HALDB or PDF partitions, Fast Path areas, change accumulations, log data sets, and RECON data sets).

4 Select types of exceptions to include:
- To view all exception types, select **All Types**.
- To view specific exception types, select **Specific**, and select all of the types that you want to include.

---

**Tip**
To display RECON exceptions (and no others), select the **RECON Data Sets** option and the **All Recovery** option.

5 Specify whether to display the Filter for exception list dialog automatically:
- To display the dialog automatically when opening the Exception List window, select the **Show filter dialog initially** check box.
- To display the dialog only when you click the **Filter** button, clear the check box.

6 Specify whether to save your filter selections:
- To use these filter selections whenever you open the Exception List window, select **Save Filter selections for future sessions** check box.
- To use these filter selections only for the current view of the Exception List window, and discard them when closing the window, clear the check box.

7 Click **OK**.

A refresh occurs automatically.
Viewing exceptions for an object

To view individual exceptions that have been reported for an object, select the exception entry for that object in the Summary table of the Exception List window.

Individual exceptions for that object are displayed in the Detail table of the Exception List. For more information about the types of exceptions that can be displayed, see “Exception types” on page 296.

You can also correct selected exceptions. For more information, see “Choosing a solution to correct selected exceptions” on page 335.

For more information about associated thresholds, see “Customizing parameters and schedules” on page 225.

Acting on exceptions for an object

To act on exceptions, you can perform one of the following actions:

- You can correct an exception immediately by using the recommended solution shortcut.
  For more information, see “Choosing the recommended solution” on page 334.

- You can select certain exceptions from a list of all available solutions to correct them.
  For more information, see “Choosing a solution to correct selected exceptions” on page 335.

- You can analyze an exception to determine why it was reported on the Exception List.
  For more information, see “Analyzing Database Advisor exceptions” on page 162.

Accessing details about objects on the Exception List

The Exception List has research capabilities so that you can access details about the database objects (databases, partitions, areas, and data sets), log objects, change accumulation objects, and RECON data sets that are reported on the Exception List.

Each row in the Details table has a pop-up menu. For database objects, the menu offers the same commands for an object that are offered when you right-click the object in the Navigation window. For example, if an exception applies to a data set, the pop-up menu shows commands that you can use to view current and history reports, view and reset base values, and view or edit parameters.
To access details about objects on the Exception List

1 To access the pop-up menu for an entry in the Exception List, right-click the entry row.

For more information about accessing detailed information from the Exception List, see “Analyzing Database Advisor exceptions” on page 162.

Exporting exception data

To export exception data for analysis outside of the console (such as in a third-party spreadsheet program), perform the following steps:

1 Click Export.

The Export Data dialog box (Figure 49 on page 309) is displayed.

Figure 49: Export Data dialog box

2 In the Select File Type field, select the type of file to write:
   - csv—Write a standard comma-separated file (fields are separated by commas).
   - txt—Write a standard tab-delimited variables file (fields are separated by tabs).

3 In the File Name field, enter the location and name of the file to write. Or you can click Browse to navigate to the location and type the file name.

4 Click Export.
Receiving notification about exceptions

Database Advisor and Recovery Advisor can notify you automatically (by sending an e-mail message, a WTO message, or both) that an exception was detected or that the status of a previously detected exception has changed.

Recipients and types of notification messages

Persons or groups of persons can be notified for different statuses of exceptions. E-mail messages, write-to-operator (WTO) messages, or both types of messages can be produced.

Frequency of notification messages

E-mail notification messages can be sent immediately as follows:

- Database Advisor can send separate email notification messages for each database exception as it is detected.

- Recovery Advisor can send an email notification message that reports all recovery and RECON exceptions that were detected during execution of Recovery Collect.

Instead of sending email notification messages immediately, Database Advisor and Recovery Advisor email messages can be consolidated into a single message per email address that is sent on a scheduled basis. If you enable consolidated messages, notification about exceptions that have a dead status can be sent immediately as well as sent in a consolidated message.

Controlling notification

To control notification, perform the following tasks:

1. Configure email/WTO parameters (which are listed in “E-mail / WTO / REXX EXEC parameters” on page 253) in the Maintain Parameters window.

   For more information, see “Editing parameters” on page 229.

2. Configure and enable the E-MAIL CONSOLIDATION schedule in the Schedules window.

   For more information, see “Editing schedules” on page 285.
Generating the Database Exception batch report

The Database Exception batch report is a list of existing exceptions of interest, as selected through specified filters. The report provides the same information as you can view in the Exception List window in the Database Management Console.

You use the EXCEPLST command of the CPCBATCH utility to request the report. The utility writes the report to a member of a specified partitioned data set, from which you can view and analyze the data.

To generate the Database Exception batch report

1. Create JCL to run the EXCEPLST command of the CPCBATCH utility.

   For more information, see “JCL for the EXCEPLST command” on page 311.

2. Specify control statements for the EXCEPLST command.

   These control statements identify the inputs and outputs of the process, and options and filters to use. For more information, see “Control Statements for the EXCEPLST command” on page 312.

3. Submit the JCL for execution.

4. Check the results of the job.

   For more information, see “Return codes for the EXCEPLST command” on page 321.

5. View the output.

   For more information, see “Database Exception report” on page 321.

JCL for the EXCEPLST command

The following figure shows example JCL to execute the EXCEPLST command of the CPCBATCH utility.

Table 34 on page 312 describes the valid statements in this JCL.

Figure 50: JCL for the EXCEPLST command

```plaintext
//CPCBATCH JOB (acct),'EXCEPTION LIST',MSGCLASS=X,REGION=0M,
//         CLASS=A,NOTIFY=&SYSUID
```
Table 34: JCL for the EXCEPLST command

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB</td>
<td><em>(Required)</em> Specify your standard JOB statement.</td>
</tr>
<tr>
<td>EXEC</td>
<td><em>(Required)</em> Specify CPCBATCH as the program to be executed. Specify the parameter string 'DLIABXPL,EXCEPLST' to execute the EXCEPLST command.</td>
</tr>
<tr>
<td>STEPLIB DD</td>
<td><em>(Required)</em> Include the library or libraries that contain your BMC global options modules and the MAXM Database Advisor and Cross-Product Connectivity (CPC) executable load modules.</td>
</tr>
<tr>
<td>outfile DD</td>
<td><em>(Optional)</em> Identify the data set to contain the Database Exception report. The ddname that you specify for this statement is the value of the OUTFILE keyword in the control statement data set. If you omit this statement, the OUTDATASET keyword is required. Use the following attributes when allocating the output data set:</td>
</tr>
<tr>
<td></td>
<td>■ Organization: PS or PO</td>
</tr>
<tr>
<td></td>
<td>■ Record format: FB</td>
</tr>
<tr>
<td></td>
<td>■ Record length: 292</td>
</tr>
<tr>
<td></td>
<td>■ Block size: 24820</td>
</tr>
<tr>
<td>CPCSYSIN DD</td>
<td><em>(Required)</em> Identify the data set that contains the EXCEPLST control statements. Specify CPCSYSIN DD * to include the control statements in the JCL.</td>
</tr>
</tbody>
</table>

Control Statements for the EXCEPLST command

The CPCSYSIN control statement data set contains 80-character fixed-length records that control the actions of the CPCBATCH utility.

**CPCSYSIN syntax**

In the CPCSYSIN data set, you can specify control statements by using the following syntax elements:
<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commands</td>
<td>The control statement must begin with a valid command. A command can have keywords and comments, separated where necessary with separators and continuation characters (if statements continue to the next line). A separator must follow a command. You can enter commands anywhere in positions 1 through 72 of the input statement (positions 73 through 80 are ignored).</td>
</tr>
<tr>
<td>Keywords</td>
<td>Keywords follow a command and invoke options. All keywords are non-positional. You specify a keyword with a value following it in parentheses. The keyword value can be any character string up to 255 characters. All alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>Comments</td>
<td>Comments consist of an alphanumeric character string beginning with a slash-asterisk (/<em>) and ending with an asterisk-slash (</em>/). Comments cannot start in position 1 of an input statement.</td>
</tr>
<tr>
<td>Separators</td>
<td>When you require a separator, use a blank, a comma, or a comment. More than one separator is allowed between keywords. Do not use a separator between a keyword and its value.</td>
</tr>
<tr>
<td>Continuation</td>
<td>Valid continuation characters are the plus sign (+) and the minus sign (-). Use them to continue control statements and comments that do not fit on a single line of input. The continuation character must be the last nonblank character. Use the continuation characters as follows:</td>
</tr>
<tr>
<td></td>
<td>■ Use the plus sign (with no spaces before it) to continue values for a single keyword to a second line (or subsequent lines). The plus sign deletes leading separators from the continued line.</td>
</tr>
<tr>
<td></td>
<td>■ Use the minus sign (with a space before it) to continue a list of keywords for a single command. The minus sign does not delete leading separators from the continued line.</td>
</tr>
</tbody>
</table>

**Keywords for the EXCEPLST command**

Figure 51 on page 313 shows all available keywords for the EXCEPLST command.

**Figure 51: Control statements for the EXCEPLST command**

```plaintext
//CPCSYSIN DD *
EXCEPLST -
OUTDS(outputDataSetName) or ODS(outputDataSetName) -
/* OUTFILE(ddname) or OFILE(ddname) - do not specify with OUTDS */ -
RECON_DSN(reconDataSetName) -
SEVLEVEL(severity) -
EXCEPTIONS(ALL or id,id,id,id,id) -
SPLITREPORT(N or Y) -
NEWEX(N or Y) -
/* The following keywords are object-type filters. Specify one or */ -
/* more keywords to include these object types and omit all others. */ -
/* Omit all of these keywords to include all object types. */ -
FULLFUNCTION(Y) -
HALDB(Y) -
FASTPATH(Y) -
CHANGEACCUM(Y) -
```
The following keywords are exception-type filters. Specify one or */ -
/* more keywords to include these exception types and omit all others. */ -
/* Omit all of these keywords to include all exception types. */ -
SPACEONLY(Y) -
SPACEANDPERF(Y) -
CADDGROUPS(Y) -
RECOVERYCONDS(Y) -
RECOVERYASSETS(Y) -
ALLRECOVERY(Y)
/*

Table 35 on page 314 describes the valid keywords for the EXCEPLST command.
For more information about keywords that filter the data by exception type or ID,
see “Exception IDs and keywords for filtering by exception type” on page 318.

Table 35: Keywords for the EXCEPLST command

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Valid Values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON_DSN</td>
<td>Data set name (maximum 44 characters)</td>
<td>None</td>
<td><em>(Required)</em> Specify the data set name of a RECON data set to read.</td>
</tr>
<tr>
<td>OUTDATASET or OUTHLQ or ODS or OHLQ</td>
<td>Data set name (maximum 44 characters)</td>
<td>None</td>
<td><em>(Required if OUTFILE is not specified)</em> Specify the data set name of the output data set to contain the Database Exception report. Use the following attributes when allocating the output data set: Organization: PS Record format: FB Record length: 292 Block size: 24820 Do not specify OUTDATASET if you specify OUTFILE.</td>
</tr>
<tr>
<td>OUTFILE or OFILE</td>
<td>DDname (maximum 8 characters)</td>
<td>None</td>
<td><em>(Required if OUTDATASET is not specified)</em> Specify the ddname of the output data set to contain the Database Exception report. Do not specify OUTFILE if you specify OUTDATASET.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Valid Values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------</td>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SEVLEVEL</td>
<td>■ 1, GOOD, or GREEN</td>
<td>3, CRITICAL, RED</td>
<td><em>(Optional)</em> Filter the data to include severity levels greater than or equal to the specified value.</td>
</tr>
<tr>
<td></td>
<td>■ 2, WARNING, or YELLOW</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ 3, CRITICAL, or RED</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ 4, DEAD, or BLACK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCEPTIONS or EXCID</td>
<td>■ ALL</td>
<td>ALL</td>
<td><em>(Optional)</em> Filter the data to include specific exception IDs.</td>
</tr>
<tr>
<td></td>
<td>■ One through five 2-digit exception IDs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPLITREPORT</td>
<td>■ N</td>
<td>N</td>
<td><em>(Optional)</em> Split the Database Exception report into four data sets. You might want to use this option if the report will contain a large amount of data.</td>
</tr>
<tr>
<td></td>
<td>■ Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NEWEX</td>
<td>■ N</td>
<td>N</td>
<td><em>(Optional)</em> Report new exceptions in a separate part of the Database Exception report.</td>
</tr>
<tr>
<td></td>
<td>■ Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FULLFUNCTION</td>
<td>Y</td>
<td>Include all object types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to full-function databases.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type.</td>
</tr>
<tr>
<td>HALDB</td>
<td>Y</td>
<td>Include all object types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to high availability large databases (HALDBs).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Valid Values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
<td>------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| FASTPATH      | Y            | Include all object types | *(Optional) Filter the data to include exceptions related to Fast Path databases.*  
If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type. |
| CHANGEACCUM   | Y            | Include all object types | *(Optional) Filter the data to include exceptions related to change accumulations.*  
If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type. |
| LOGDATASETS   | Y            | Include all object types | *(Optional) Filter the data to include exceptions related to log data sets.*  
If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type. |
| RECONS        | Y            | Include all object types | *(Optional) Filter the data to include exceptions related to RECON data sets.*  
If you specify this object-type filter, the Database Exception report omits exceptions for all other object types unless you specify a corresponding filter for that object type. |
| SPACEONLY     | Y            | Include all exception types | *(Optional) Filter the data to include exceptions related to space usage.*  
If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type. |
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Valid Values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPACEANDPERF</td>
<td>Y</td>
<td>Include all exception types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to space usage and performance. If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type.</td>
</tr>
<tr>
<td>CADBDSGROUPS</td>
<td>Y</td>
<td>Include all exception types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to change accumulation groups or database data set groups. If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type.</td>
</tr>
<tr>
<td>RECOVERYCONDS</td>
<td>Y</td>
<td>Include all exception types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to recovery conditions. If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type.</td>
</tr>
<tr>
<td>RECOVERYASSETS</td>
<td>Y</td>
<td>Include all exception types</td>
<td><em>(Optional)</em> Filter the data to include exceptions related to recovery assets. If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type.</td>
</tr>
<tr>
<td>ALLRECOVERY</td>
<td>Y</td>
<td>Include all exception types</td>
<td><em>(Optional)</em> Filter the data to include all recovery exceptions. If you specify this exception-type filter, the Database Exception report omits exceptions for all other exception types unless you specify a corresponding filter for that exception type.</td>
</tr>
</tbody>
</table>
Exception IDs and keywords for filtering by exception type

The following table lists the IDs that you can use as values for the EXCEPTIONS keyword to include the corresponding exception in the Database Exception report.

The table also indicates the keywords that you can use to include the corresponding exception when you filter the data by exception type.

Table 36: Exception IDs and keywords for filtering by exception type

<table>
<thead>
<tr>
<th>Exception</th>
<th>ID</th>
<th>Keyword for filtering by exception type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYZE FOR DBD OVERDUE</td>
<td>99</td>
<td>None</td>
</tr>
<tr>
<td>AREA DOVF UTILIZATION TOO HIGH</td>
<td>21</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>AREA FRAGMENTATION TOO HIGH</td>
<td>18</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>AREA I/O PROBABILITY TOO HIGH</td>
<td>19</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>AREA IOVF SPACE LOW</td>
<td>15, 16</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>AREA RAA UTILIZATION TOO HIGH</td>
<td>20</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>AREA SDEP FREE SPACE TOO LOW</td>
<td>17</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>BACKOUT NEEDED</td>
<td>A0</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>CA DATA SET NOT CATALOGED</td>
<td>C9</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>CA RUN IN ERROR</td>
<td>D0</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>COLLECT FOR DBD OVERDUE</td>
<td>98</td>
<td>None</td>
</tr>
<tr>
<td>CPC COLLECT TASK NOT SCHEDULED</td>
<td>97</td>
<td>None</td>
</tr>
<tr>
<td>DBDS IN MULTIPLE DBDS GROUPS</td>
<td>C1</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>DBDS NOT IN ANY DBDS GROUP</td>
<td>C2</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>DBDS NOT IN CHANGE ACCUM GROUP</td>
<td>C3</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>DSN SIZE EXCEEDS THRESHOLD</td>
<td>E9</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>EEQE(S) FOUND</td>
<td>A1</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>GENMAX BELOW THRESHOLD</td>
<td>B0</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>GENMAX EXCEEDED THRESHOLD</td>
<td>B2</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>Exception</td>
<td>ID</td>
<td>Keyword for filtering by exception type</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>----</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>GRPMAX BELOW THRESHOLD</td>
<td>C6</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>GRPMAX EXCEEDED THRESHOLD</td>
<td>C8</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>HALDB PARTITION NOT INITIALIZED</td>
<td>A5</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>HDAM RAA DEFINITION TOO LARGE</td>
<td>6</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>HDAM RAA DEFINITION TOO SMALL</td>
<td>5</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>I/O PROBABILITY TOO HIGH</td>
<td>1</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>IC DATA SET IN ERROR</td>
<td>C5</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>IC DATA SET NOT CATALOGED</td>
<td>C4</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>IC RECOMMENDED BY CHANGE ACCUM</td>
<td>F0</td>
<td>None</td>
</tr>
<tr>
<td>IMAGE COPY NEEDED</td>
<td>A3</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>IMAGE COPY RECOMMENDED</td>
<td>A4</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>INSUFF ALLOC FREE SPACE (LS)</td>
<td>35</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFF TOTAL FREE SPACE (LS)</td>
<td>34</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFFICIENT ALLOC FREE SPACE</td>
<td>33</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFFICIENT FREE BLOCKS</td>
<td>29</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFFICIENT FREE SPACE</td>
<td>E6</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>INSUFFICIENT FREE SPACE IN RAA</td>
<td>8</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFFICIENT SPACE</td>
<td>28</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>INSUFFICIENT TOTAL FREE SPACE</td>
<td>32</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>MIN # OF CA RUNS NOT AVAILABLE</td>
<td>D2</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>MIN # OF IC’S NOT AVAILABLE</td>
<td>C0</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>MORE LOGS THAN THRESHOLD</td>
<td>B9</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO AREA DATA SET AVAILABLE</td>
<td>A8</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO AREA DATA SET DEFINED</td>
<td>A9</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO CA RUNS WITHIN TIME RANGE</td>
<td>D1</td>
<td>CADBDSGROUPS</td>
</tr>
<tr>
<td>NO IC WITHIN TIME RANGE</td>
<td>B7</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO SECIC WITHIN TIME RANGE</td>
<td>B8</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO SECLOG WITHIN TIME RANGE</td>
<td>E1</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO SECSDL WITHIN TIME RANGE</td>
<td>E2</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>NO VALID IMAGE COPY</td>
<td>B6</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>Exception</td>
<td>ID</td>
<td>Keyword for filtering by exception type</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>----</td>
<td>----------------------------------------------------------------</td>
</tr>
<tr>
<td>PREOPEN NOT SET FOR AREA</td>
<td>F1</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>PRILOG ENTRY IN ERROR</td>
<td>D7</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>PRILOG ENTRY NOT CATALOGED</td>
<td>D3</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>PRILOG ENTRY IN ERROR</td>
<td>D9</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>PRILOG ENTRY NOT CATALOGED</td>
<td>D5</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>PROHIBIT AUTHORIZATION FOUND</td>
<td>A7</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>RAA TOTAL FREE SPACE TOO SMALL</td>
<td>9</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>READ ONLY</td>
<td>A6</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>RECON DISCARDED/UNAVAILABLE</td>
<td>E3</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>RECOVERY NEEDED</td>
<td>A2</td>
<td>RECOVERYCONDS, ALLRECOVERY</td>
</tr>
<tr>
<td>RECOVPD BELOW THRESHOLD</td>
<td>B3</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>RECOVPD EXCEEDED THRESHOLD</td>
<td>B5</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>SECLOG ENTRY IN ERROR</td>
<td>D8</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>SECLOG ENTRY NOT CATALOGED</td>
<td>D4</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>SECSDLD ENTRY IN ERROR</td>
<td>E0</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>SECSDLD ENTRY NOT CATALOGED</td>
<td>D6</td>
<td>RECOVERYASSETS, ALLRECOVERY</td>
</tr>
<tr>
<td>TOO MANY CA SPLITS</td>
<td>31</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY CA SPLITS</td>
<td>E8</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>TOO MANY CI SPLITS</td>
<td>30</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY CI SPLITS</td>
<td>E7</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>TOO MANY EXTENTS</td>
<td>26</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY EXTENTS</td>
<td>E4</td>
<td>ALLRECOVERY</td>
</tr>
<tr>
<td>TOO MANY FRAGMENTED DB RECORDS</td>
<td>3</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY ROOTS IN OVERFLOW</td>
<td>7</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY SEGMENTS SPLITS</td>
<td>2</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY SPLITS ACROSS BLOCKS</td>
<td>4</td>
<td>SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY VOLUMES</td>
<td>E5</td>
<td>SPACEONLY, SPACEANDPERF</td>
</tr>
<tr>
<td>TOO MANY VOLUMES</td>
<td>27</td>
<td>ALLRECOVERY</td>
</tr>
</tbody>
</table>
Return codes for the EXCEPLST command

The following table explains the return codes that the EXCEPLST command can issue.

<table>
<thead>
<tr>
<th>Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The function completed successfully.</td>
</tr>
<tr>
<td>8</td>
<td>A CPC connection error occurred.</td>
</tr>
<tr>
<td>12</td>
<td>No output data set was defined.</td>
</tr>
<tr>
<td>16</td>
<td>One or more control statement syntax errors were detected.</td>
</tr>
</tbody>
</table>

Database Exception report

The following figures show an example of the Database Exception report. (The report is wider than can be displayed in a single view.)

**Figure 52: Database Exception report (left view)**

```
*** DATABASE EXCEPTION REPORT *** SEVERITY LEVEL: WARNING  EXCEPTION ID #S: ALL
STATUS ,HOST/SYSPLEX, IMSPLEX NAME , OBJ TYPE ,OBJ NAME,
WARNING DATE ,CRITICAL DATE,DEAD DATE , DBD ,HIOFM14
CRITICAL,MXA4 ,MXA EXCEPTION REGRESSION , .DBD , .HIOFM14
```

**Figure 53: Database Exception report (middle view)**

```
WARNING DATE ,CRITICAL DATE,DEAD DATE ,DBD NAME,PARTITION,DD NAME ,ID,
N/A ,10/01/2013 ,10/06/2013,HIOFM14 ,N/A      ,HIOFM143,26,
```

**Figure 54: Database Exception report (right view)**

```
DESCRIPTION ,     THRESHOLD      ,  CURRENT VALUE ,DSG ,DB TYPE ,DS TYPE
TOO MANY EXTENTS ,50                  ,10              ,3   ,HIDAM   ,OSAM
```

The title line of the report indicates the filter criteria that you specified. The next line contains column headings. Each of the remaining lines contains information about an exception.

You can download the data to a personal computer and import it into a spreadsheet program for analysis. The fields are separated by commas.
Working with solutions, JCL generation, and viewing jobs

This section contains information about using solutions to correct reported exceptions, generating JCL by using Advisor functions and utilities, and viewing jobs and their output.

Overview of Advisor solutions and JCL generation

This section explains how Advisor solutions are defined and how they are used during JCL generation.

Overview of the solution process

The solution process fulfills the execute portion of the Advisor database maintenance cycle.

The process recommends the best solutions to correct problems that are reported by Database Advisor and Recovery Advisor collection and analysis processes. You can correct all (or certain) exceptions by using the recommended solutions or any available solutions.

Note

The processes that Recovery Advisor uses to recommend solutions and generate JCL also apply to Database Advisor. If Recovery Advisor and Database Advisor report exceptions for the same object, the solution process chooses the best solutions to correct exceptions.

When you select a solution, the Generate JCL function generates JCL that is optimized for your environment. When the generated JCL is executed, the problems that are indicated by the exceptions are corrected.
Parameters that control solution processing include goal parameters, job control parameters, and parameters for specific functions such as the Image Copy function and the Unload function.

**Determining which solutions are available**

A solutions table contains a list of BMC utilities, IBM utilities, and standard IMS industry utilities.

The Solution process detects the solutions that you are licensed to use and filters your list of available solutions, based on availability.

Each solution on your list has a solution definition that identifies the attributes of the solution, such as whether the solution can be used with VSAM data sets or whether the solution requires that a database be taken offline. The Solution process uses solution attributes to determine a solution recommendation.

For more information about the solution list, see “Maintaining solutions” on page 379.

**Matching solutions with your goals**

Goals are parameter values that establish space, performance, and availability priorities.

Space, performance, and availability goals are considered when the Solution process is determining the best available solutions to recommend.

For example, if you specify the database availability goal Can take offline briefly, the Solution process recommends solutions that require only a brief outage.

**Determining solution recommendations**

The Solution process recommends the best solutions to correct a single exception or a combination of exceptions.

The recommendation is based on the reported problems, the solutions that you are licensed to use, and goal parameters.

The Solution process establishes relationships among the solutions. Solution relationships can be inclusive, exclusive, or equivalent.
Example

An example of an inclusive solution relationship would be a reorganization and an image copy. Reorganization includes an image copy, so it is unnecessary to run both jobs. You can run the reorganization and solve both problems.

An example of an exclusive solution relationship would be a reorganization and a recovery. If you run a reorganization and then run a recovery, you lose the results of the reorganization.

An example of an equivalent solution relationship would be a single-step reorganization and a multi-step reorganization. You can choose single-step reorganization or multi-step reorganization and achieve the same results.

When an exception is reported to the Exception List and you click the Solutions button, the Solution process identifies all possible solutions that can be used to correct all reported problems for the database. Then the process looks at the solution relationships and the goal parameters to identify the best solutions to execute.

For more information about recommended solutions, see “Using solutions from the Exception List” on page 333.

Optimizing JCL

When you select a recommended solution or select another available solution, you can request that the Generate JCL function create JCL to execute the solution.

The function generates JCL that is optimized for your environment by examining job control parameters and other parameters for specific functions and utilities (such as Image Copy or Unload functions).

Suggested default values are provided for most JCL parameters at the host level. However, you must specify a few site-specific values so that the Generate JCL function can make the best decisions and recommendations for your site. For information about required JCL parameters, see “Parameters that influence JCL generation” on page 332.

Overview of Advisor solutions

You can use Advisor solutions to handle reported exceptions and automate tasks.

A solution is used to generate JCL for executing a utility or function to accomplish the task, such as reorganizing or recovering a database. BMC provides solutions for most common purposes, and you can define your own solutions as required for your environment.
**Solution definitions**

An Advisor solution definition is a repository record that contains the following elements:

An Advisor solution definition contains the following elements:

- **Identifying information**
  Each solution is identified by a unique combination of solution name, feature name, class, and type (BMC or USER).

- **Name of a JCL script**
  A JCL script is a list of instructions to be performed during a JCL generation process that uses this solution. For example, a script can obtain parameter values from the Advisor repositories to build a JOB statement or a control statement for a utility. A script can call another script.

  Advisor products provide many scripts to perform various tasks. You can also write your own scripts by using the JCL script language that BMC provides. For JCL script language syntax, variable processing rules, and available statements, see “Using the JCL script language” on page 499.

- **Solution rules**
  Rules define whether the solution applies to an object. For example, rules can state whether a solution applies to OSAM databases, VSAM databases, or Fast Path databases. Rules can also provide option values to use during JCL generation. Rules are defined when a solution is created.

  For more information about working with Advisor solution definitions, see “Maintaining solutions” on page 379.

**BMC-provided Advisor solutions**

BMC provides a full range of Advisor solutions to perform tasks in an IMS database administration environment.

*Table 37 on page 327* lists the BMC-provided solutions.

Each Advisor solution is identified by a unique combination of solution name, feature name, and class values. If you are specifying keywords for the JCL Generation utility, you must specify the correct combination of values for the solution that you want to use. Other types of values are also assigned to solutions. You can use the Maintain Solutions window to view these values, as explained in “Viewing solution information” on page 379.
Table 37: BMC-provided Advisor solutions and valid SOLUTION, CLASS, and FEATURE keyword combinations

<table>
<thead>
<tr>
<th>Solution (SOLUTION keyword value)</th>
<th>Feature (FEATURE keyword value)</th>
<th>Class (CLASS keyword value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYZER</td>
<td>OFFLINE DB</td>
<td>FP</td>
</tr>
<tr>
<td>ANALYZER</td>
<td>OFFLINE IC</td>
<td>FP</td>
</tr>
<tr>
<td>ANALYZER</td>
<td>ONLINE</td>
<td>FP</td>
</tr>
<tr>
<td>BATCH CMD INTERFACE</td>
<td>START DATABASE</td>
<td>BRI</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>BUILD IDCAMS DELETE/DEFINE STATEMENTS</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>CLEANUP ORPHANED REPOSITORY RECORDS</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>DISASSEMBLE IMS Control BLOCKS</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>GENERATE SOLUTIONS FOR EXCEPTIONS</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>REPOSITORY EXTRACT IN PDX FORMAT</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>REPOSITORY MAINTENANCE EXPAND DATAGROUP</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BATCH UTILITIES</td>
<td>UNREGISTER AND REMOVE REPOSITORY RECORDS</td>
<td>ADVISOR</td>
</tr>
<tr>
<td>BUILD</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
<tr>
<td>CHANGE</td>
<td>OFFLINE IC</td>
<td>FP</td>
</tr>
<tr>
<td>CHANGE ACCUM</td>
<td>CHANGE ACCUMULATION</td>
<td>BRI</td>
</tr>
<tr>
<td>DATABASE RECOVERY</td>
<td>RECOVER TO CURRENT</td>
<td>BRI</td>
</tr>
<tr>
<td>DATABASE RECOVERY</td>
<td>RECOVER TO CURRENT</td>
<td>IBM</td>
</tr>
<tr>
<td>DATABASE RECOVERY</td>
<td>RECOVER TO CURRENT BY CAGRP</td>
<td>BRI</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE CAGRP ADD</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE CAGRP GRPMAp</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDS GENMAX</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDS PREOPEN</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDS RECOVPD</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDSGRP ADD</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDSGRP REMOVE</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>LIST CAGRP</td>
<td>IBM</td>
</tr>
<tr>
<td>Solution (SOLUTION keyword value)</td>
<td>Feature (FEATURE keyword value)</td>
<td>Class (CLASS keyword value)</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>LIST DBDS</td>
<td>IBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>LIST LOG</td>
<td>IBM</td>
</tr>
<tr>
<td>EXTEND</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
<tr>
<td>EXTEND</td>
<td>ONLINE</td>
<td>FP</td>
</tr>
<tr>
<td>EXTRACT</td>
<td>OFFLINE DB</td>
<td>FP</td>
</tr>
<tr>
<td>EXTRACT</td>
<td>OFFLINE IC</td>
<td>FP</td>
</tr>
<tr>
<td>EXTRACT</td>
<td>ONLINE</td>
<td>FP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>MULTI STEP DBIL REORG USING SHADOWS</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>SINGLE STEP DBR REORG USING SHADOWS</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>INITIALIZE DATABASE</td>
<td>EP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>MULTI STEP DBIL REORG USING SHADOWS</td>
<td>EP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>REORG USING IC RESTORE</td>
<td>EP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>REORG USING UNLOAD FILE INPUT</td>
<td>EP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>SINGLE STEP DBIL REORG USING XIM</td>
<td>EP</td>
</tr>
<tr>
<td>FAST REORG</td>
<td>SINGLE STEP DBR REORG USING SHADOWS</td>
<td>EP</td>
</tr>
<tr>
<td>IDCAMS</td>
<td>ADD VOLUME</td>
<td>OTHER</td>
</tr>
<tr>
<td>IDCAMS</td>
<td>REPRO</td>
<td>OTHER</td>
</tr>
<tr>
<td>IEFBR14</td>
<td>ADD VOLUME</td>
<td>OTHER</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>BATCH IMAGE COPY</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>BATCH IMAGE COPY</td>
<td>IBM</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>BATCH/CONCURRENT IMAGE COPY</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>CONCURRENT IMAGE COPY</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>COPY IMAGE COPY</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>IC TRIGGERING IMAGE COPY</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>IMAGE COPY BY CAGRP</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>IMAGE COPY BY RMGR GROUP</td>
<td>BRI</td>
</tr>
<tr>
<td>IMAGE COPY</td>
<td>ONLINE</td>
<td>FP</td>
</tr>
<tr>
<td>Solution (SOLUTION keyword value)</td>
<td>Feature (FEATURE keyword value)</td>
<td>Class (CLASS keyword value)</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD ALL</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD PRIMARY INDEX</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD SECONDARY INDEXES</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD ALL</td>
<td>EP</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD PRIMARY INDEX</td>
<td>EP</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>BUILD SECONDARY INDEXES</td>
<td>EP</td>
</tr>
<tr>
<td>INDEX BUILD</td>
<td>REBUILD ILDS</td>
<td>EP</td>
</tr>
<tr>
<td>INITIALIZE</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
<tr>
<td>INITIALIZE</td>
<td>OFFLINE</td>
<td>FP_IBM</td>
</tr>
<tr>
<td>ONLINE DEFRAG</td>
<td>USING BLOCK MODE</td>
<td>EP</td>
</tr>
<tr>
<td>ONLINE DEFRAG</td>
<td>USING RECORD MODE</td>
<td>EP</td>
</tr>
<tr>
<td>ONLINE DEFRAG</td>
<td>USING SCAN MODE</td>
<td>EP</td>
</tr>
<tr>
<td>ONLINE REORG</td>
<td>DB COPY OR INDEX REORG</td>
<td>EP</td>
</tr>
<tr>
<td>ONLINE REORG</td>
<td>REORG USING SHADOWS</td>
<td>EP</td>
</tr>
<tr>
<td>POINTER CHECK</td>
<td>FULL CHECK DB</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>POINTER CHECK</td>
<td>FULL CHECK IC</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>POINTER CHECK</td>
<td>HASH CHECK DB</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>POINTER CHECK</td>
<td>HASH CHECK IC</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>RECON UTILITIES</td>
<td>RECON CLEANUP FOR DISASTER RECOVERY</td>
<td>BRI</td>
</tr>
<tr>
<td>RECON UTILITIES</td>
<td>REORG ALL RECON DATA SETS</td>
<td>BRI</td>
</tr>
<tr>
<td>RECON UTILITIES</td>
<td>REPLACE RECON DATA SET</td>
<td>BRI</td>
</tr>
<tr>
<td>RECON UTILITIES</td>
<td>LIST RECON STATUS</td>
<td>IBM</td>
</tr>
<tr>
<td>RELOAD</td>
<td>INITIALIZE DATABASE</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>RELOAD</td>
<td>INITIALIZE DATABASE</td>
<td>EP</td>
</tr>
<tr>
<td>RELOAD</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
<tr>
<td>REORG</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
<tr>
<td>REORG</td>
<td>ONLINE</td>
<td>FP</td>
</tr>
<tr>
<td>REORG</td>
<td>ONLINE</td>
<td>FP_IBM</td>
</tr>
<tr>
<td>SDEP UTILITIES</td>
<td>SDEP DELETE</td>
<td>FP_IBM</td>
</tr>
<tr>
<td>SDEP UTILITIES</td>
<td>SDEP SCAN</td>
<td>FP_IBM</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBIL REORG</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>Solution (SOLUTION keyword value)</td>
<td>Feature (FEATURE keyword value)</td>
<td>Class (CLASS keyword value)</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>---------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBR REORG</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBIL REORG</td>
<td>EP</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBR REORG</td>
<td>EP</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBIL REORG</td>
<td>IBM</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBR REORG</td>
<td>IBM</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBIL REORG</td>
<td>MIX</td>
</tr>
<tr>
<td>STD REORG</td>
<td>MULTI STEP DBR REORG</td>
<td>MIX</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>UNLOAD</td>
<td>CLASSIC</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>PARALLEL UNLOAD</td>
<td>EP</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>UNLOAD</td>
<td>EP</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>OFFLINE DB</td>
<td>FP</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>OFFLINE IC</td>
<td>FP</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>UNLOAD</td>
<td>IBM</td>
</tr>
<tr>
<td>UNLOAD/RELOAD</td>
<td>OFFLINE</td>
<td>FP</td>
</tr>
</tbody>
</table>

**User-defined solutions**

You can create your own solution definitions.

Your user-defined solutions can use any combination of BMC-provided JCL scripts and scripts that you write yourself by using the JCL script language that BMC provides. For more information about adding your own solution definitions, see “Maintaining solutions” on page 379. For more information about the script language, see “Using the JCL script language” on page 499.

**Overview of JCL generation**

The JCL generation process executes a script that is named in a selected solution and creates JCL for a selected object.

The JCL generation process obtains option values from solution rules and from certain parameters that you have defined in the Maintain Parameters window. For example, the Job Control category of parameters defines the values to be used to create the JOB statement, to set the region size, to allocate work files, and so on. You can also customize parameter values for functions and utilities, such as Image Copy and Unload. For more information, see “Parameters that influence JCL generation” on page 332.
To generate appropriate JCL, the Generate JCL function dynamically obtains information about the selected object (for example, the database type: full-function, HALDB, or Fast Path) and about the BMC products and other functions and utilities that are available for use in the environment.

Methods for working with Advisor solutions and generated JCL

You can use the following methods to work with Advisor solutions and generated JCL:

- You can view definitions of BMC-provided solutions and user-defined solutions that are available in your environment. You can also add, delete, and edit user-defined solutions. For more information, see “Maintaining solutions” on page 379.

- You can use solutions directly from the Exception List window, which lists objects for which one or more exceptions have been reported. You can select an object in the Exception List window and choose a recommended solution to examine a list of solutions to correct individual exceptions that were reported for the object. For more information, see “Using solutions from the Exception List” on page 333.

- You can use solutions to correct exceptions or perform tasks on demand by using the Generate JCL function. To access the Generate JCL function, you select an object from the object tree in the Navigation window and select **Generate JCL** from the right-click menu. For more information, see “Generating JCL from the Generate JCL window” on page 338.

- You can use the Solution utility and the JCL Generation utility to handle exceptions in batch rather than interactively through the Exception List window or Generate JCL window. The Solution utility obtains a list of reported exceptions for the objects that you select, determines the recommended solutions to correct the exceptions, and generates control statements that can be passed to the JCL Generation utility. The JCL Generation utility generates JCL to execute the recommended solutions. You can run these utilities on a scheduled basis for full automation of exception handling. For more information, see “Handling exceptions with the Solution batch utility” on page 341 and “Generating JCL with the JCL Generation utility” on page 350.

**Note**

You can use the Generate JCL function to create JCL for the Solution utility and the JCL Generation utility. Right-click a RECON or database group object in the navigation window, choose **Generate JCL**, select the row that contains GENERATE SOLUTIONS FOR EXCEPTIONS, and click Generate JCL.
You can use the Jobs perspective or the JES viewer in the console. The Jobs perspective and the JES viewer provide a convenient interface for tracking and displaying jobs in your environment, including jobs that have been generated by Advisor functions and utilities. For more information, see “Viewing jobs in the console” on page 372.

Parameters that influence JCL generation

The following categories of user-controlled parameters influence JCL generation:
- Job Control
- Unload
- Reorg
- Image Copy
- Additional Fast Path
- Repository Management (for Fast Path)
- Recovery & Change Accum

Most of the parameters that are relevant to the Generate JCL function and BMC-provided solutions have default values. However, the following parameters are required for all or some solutions but do not have default values:
- Job Card Statement
- Work File UNIT Name
- Work File High Level Qualifier
- Sort Work UNIT Name
- IDCAMS Delete Control Statements
- IDCAMS Define Control Statements

You must provide values for these parameters before you generate JCL for a solution that requires these parameters. You can edit the parameters at various levels in the object tree. The defaults are populated at the host level and are inherited by all subordinate objects until you customize the parameters.

For information about how to specify parameter values, see Customizing parameters on page 226.

User-defined DD statements in generated JCL

You can define custom DD statements to be included in the JCL that is generated by automated Advisor processes, BMC-provided JCL scripts, or your own JCL scripts.
Define custom DD statements by using the DD Names parameters in the Job Control category in the Maintain Parameters window. For more information, see “Job Control parameters” on page 260.

Table 38 on page 333 describes DD statements that are used for specific purposes in automated Advisor processes and BMC-provided JCL scripts.

### Table 38: DD statements that are used for specific purposes

<table>
<thead>
<tr>
<th>DD statement</th>
<th>Description</th>
</tr>
</thead>
</table>
| BMCPSWD           | Identifies the BMC password library  
If you define this DD statement, most generated jobs (including utility JCL generations and external tasks that are scheduled for the CPC subsystem) include this statement. |
| RCUPARMS          | Identifies a library or library concatenation that contains your customized global options modules and predefined dynamic allocation models          |
| REPGDGBK or REPBKUP | Defines the data set name to contain repository data that is written by the scheduled CPC repository backup utility  
  - The REPGDGBK statement identifies a GDG data set. You must also define the GDG model to which this statement refers.  
  - The REPBKUP statement identifies a non-GDG output data set.  
If you define this DD statement, define it at the host (SYSPLEX) level.  
For more information about how the utility uses these statements, see the Database Products for IMS Customization Guide. |

**Using solutions from the Exception List**

After an exception has been reported for an object in the Exception List window, you can correct the exception by selecting the object and generating and submitting a solution.

The Advisor Solution process helps you to implement consistent practices by recommending the best solution for an exception or a combination of exceptions. Given the same conditions, the process produces the same recommendation every time. Solution recommendations are based on the following factors:

- Actual or predicted problem
- Solutions that are available to correct the problem
- User-defined goals
The solution process considers all factors and provides a list of solutions that you can execute.

*Note*

In some situations, the recommended solution does not solve all exceptions but instead solves the exception that is perceived to be the most important exception. For example, if the Recovery Needed exception and the GENMAX Below Threshold exception are detected for a database data set, the recommended solution is to recover the database.

To obtain solutions and generate JCL, you can use any of the following approaches:

- Choose the recommended solution
- Choose a solution to correct selected exceptions
- Use the recommended JCL shortcut

### Choosing the recommended solution

Use the following procedure to obtain the recommended solution.

1. In the Exception List window, select a row in the Summary table.
2. Click Solutions.

   The Solutions tab (Figure 55 on page 334) in the Solutions window is displayed.

**Figure 55: Solutions tab (Solutions window) showing recommended solution only**

The Solutions window has an Exceptions area (left) and a Solutions area (right). The default view of the Exceptions area shows the selection to correct all
exceptions that have been reported for the database. The default view for the Solutions area shows the recommended solution to correct all exceptions.

3 Click Generate JCL.

JCL is generated and displayed on the JCL tab (“Using the recommended JCL shortcut” on page 337).

4 Perform one of the following steps:
   ■ To edit the JCL, click Edit.
   ■ To submit the JCL, click Submit.
   ■ To save the JCL, click Save to Mainframe.
   ■ To clear the JCL, click Clear.

For more instructions, see “Working with generated JCL” on page 339.

Choosing a solution to correct selected exceptions

Instead of choosing the recommended solution or choosing to correct all exceptions for an object, you can choose from a list of available solutions to correct selected exceptions for the object.

1 In the Exception List window, select a row in the Summary table.

2 Click Solutions.

   The Solutions tab in the Solutions window is displayed.

3 In the Exceptions area, select the Fix Selected Exceptions option and select the check box for each exception that you want to correct.

4 Click Find Solutions.
The available solutions to correct the selected exceptions are listed in the Solutions area on the Solutions tab (Figure 56 on page 336).

5 In the Exceptions area, select **Fix Selected Exceptions** and select one or more exceptions.

6 In the Solutions area, select a solution.

   The default is to use the common solutions.

7 Click **Generate JCL**.

   JCL is generated and displayed on the JCL tab (Figure 57 on page 337).

8 Perform one of the following steps:
   ■ To edit the JCL, click **Edit**.
   ■ To submit the JCL, click **Submit**.
   ■ To save the JCL, click **Save to Mainframe**.
   ■ To clear the JCL, click **Clear**.

   For more instructions, see “Working with generated JCL” on page 339.
Using the recommended JCL shortcut

The quickest way to correct all reported exceptions for an object is to use the recommended JCL shortcut. You can use the shortcut directly from the Exception List. The shortcut generates the recommended JCL to correct exceptions that were reported for the selected object in the Summary table.

1. In the Exception List window (“Working with exceptions and the Exception List” on page 303), right-click a row in the Summary table and choose Recommended JCL.

JCL is generated and displayed on the JCL tab in the Solutions window (Figure 57 on page 337).

Figure 57: JCL tab in the Solutions window

2. Perform one of the following steps:
   - To edit the JCL, click Edit.
   - To submit the JCL, click Submit.
   - To save the JCL, click Save to Mainframe.
   - To clear the JCL, click Clear.

For more information about these steps, see “Working with generated JCL” on page 339.
Generating JCL from the Generate JCL window

This section describes how to use the Generate JCL function from the Generate JCL window to create JCL for correcting exceptions and performing other database administration tasks.

In the console, you can access the Generate JCL function from the Solutions window and from the right-click menus for RECON, group, database, partition, DEDB area, and change accumulation objects. You can edit, save, and submit the generated JCL directly from the Generate JCL function.

Generating optimized JCL

You can generate JCL for RECON, group, database, partition, DEDB area, and change accumulation group objects.

1. In the Navigation window, right-click the object for which you want to generate JCL, and choose Generate JCL.

The solutions that are available and appropriate for the object are displayed on the Solutions tab in the Generate JCL window (Figure 58 on page 338).

Figure 58: Solutions tab (Generate JCL window)

The types of solutions that are available for the selected object are displayed in the Solution Filter list.
2 In the Solution/Feature List table on the Solutions tab, select a solution and a solution feature.

3 Click Generate JCL.

Depending on the solution, the Generate JCL dialog box (Figure 59 on page 339) might be displayed to request additional parameters that are required for JCL generation.

Figure 59: Generate JCL dialog box

4 If applicable, provide values for the additional parameters and click OK.

The Generate JCL function generates the JCL and displays it on the JCL tab in the Generate JCL window (Figure 60 on page 339).

Figure 60: JCL tab (Generate JCL window)

**Working with generated JCL**

When you have generated JCL, you can edit, clear, submit, and save the JCL. You can edit JCL directly in the console.
Editing JCL

You can edit JCL, in the JCL editor.

1 On the JCL tab in the Generate JCL window ("Generating optimized JCL" on page 338), click Edit.

   The JCL editor is displayed.

2 Type your edits in the JCL editor.

3 (optional) To search the JCL to find and replace character strings, click Find/Replace.

Clearing JCL

If you have generated JCL and then return to the Solutions tab to generate different or additional JCL, you must clear the existing JCL on the JCL tab (unless you want to append additional JCL to the existing JCL). If you do not clear the existing JCL, the additional JCL that you generate is appended to the existing JCL.

1 To clear JCL from the JCL tab, click Clear.

Submitting JCL

You can submit JCL that is displayed on the JCL tab.

1 To submit JCL that is displayed on the JCL tab, click Submit.

   You can track the status of the job and view job output by using the Jobs perspective or the JES viewer. For more information about viewing jobs and their output, see “Viewing jobs in the console” on page 372.

Saving JCL

You can save JCL that is displayed on the JCL tab.

1 On the JCL tab, click Save to Mainframe.

2 In the Save to Mainframe dialog box, specify or select a partitioned data set (PDS) name and a member name.
**Note**

The dialog box saves the last five data set names and member names that you have entered. If a name that you want to use is saved, you can select the name instead of specifying it.

3. Click **Save** to store the JCL in the PDS.

**Handling exceptions with the Solution batch utility**

The Solution batch utility (program BRIUBSOL) automates the process of generating JCL for exceptions that are reported on the Exception List.

For example, if you schedule the Solution utility to run every night, optimized JCL is generated for each object with exceptions and is ready to submit the next morning. The Solution utility can process exceptions for databases, change accumulations, logs, and RECON data sets.

The Solution utility reads exceptions from the repository and builds control statements to generate JCL. The control statements can be passed to the JCL Generation utility (CPCBATCH). For more information about CPCBATCH, see “Generating JCL with the JCL Generation utility” on page 350.

The Solution utility processes all exceptions that are related to a single IMSPLEX. A separate execution of the Solution utility is required for each IMSPLEX in the environment. Each job for the Solution utility should use a different data set that is identified in the GENJCL DD statement.

**Coding JCL for the Solution utility**

The following figure shows sample JCL for the Solution utility.

Sample JCL is located in member BRIUBSOL of the product sample library.

**Figure 61: Sample Solution utility JCL**

```
//jobname  JOB <standard job statement parameters>,REGION=OM
//BRIUBSOL EXEC PGM=BRIUBSOL
//STEPLIB  DD DISP=SHR,DSN=cpc.options.library
//         DD DISP=SHR,DSN=bmc.load.library
//SYSUDUMP DD SYSOUT=*  
//LIST     DD SYSOUT=*  
//SOLOUT   DD SYSOUT=*  
//GENJCL   DD DISP=(NEW,CATLG,DELETE),UNIT=SYSDA, 
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=3120).
```
The USE command and the RECON keyword are required.

* You can code multiple DB and RSRC commands.

* If you code the DB command, the following keywords are optional:

  DBD=dbdname or mask | DBGROUP=dbgroupname +
  AREA=areaname or mask | PART=partname or mask

* NOWARN +
* ONLYNEW +
* LIST

* If you code the RSRC command, the following keywords are optional

  TYPE=CA | LOG | RCON +
  NAME=change accum group | log subsystem name | mask +

* NOWARN +
* ONLYNEW +
* LIST

Table 39 on page 342 lists DD statements that are used for the Solution utility.

Table 39: Solution utility DD statements

<table>
<thead>
<tr>
<th>DD statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB DD</td>
<td><em>(Required)</em> In the STEPLIB concatenation, include the library that contains your Cross Product Connectivity (CPC) options module and the library that contains the BMC product load modules. APF authorization is not required.</td>
</tr>
<tr>
<td>SYSUDUMP DD</td>
<td><em>(Optional)</em> Specify a SYSOUT data set to contain diagnostic information that can be written if a failure occurs.</td>
</tr>
<tr>
<td>LIST DD</td>
<td><em>(Optional)</em> Specify a SYSOUT data set to contain control statement processing information. If you omit this statement, the utility dynamically generates it.</td>
</tr>
<tr>
<td>SOLOUT DD</td>
<td><em>(Optional)</em> Specify a SYSOUT data set to contain the solution processing report. If you specify the LIST control statement keyword, the SOLOUT statement also contains a list of the GENJCL output that was produced. If you omit this statement, the utility dynamically generates it.</td>
</tr>
<tr>
<td>GENJCL DD</td>
<td><em>(Required)</em> This statement must specify an output file with a logical record length (LRECL) of 80 or a SYSOUT data set. This output file contains GENJCL statements that are written by the Solution utility and that are used as input to the JCL Generation utility (CPCBATCH).</td>
</tr>
<tr>
<td>SOLVE DD</td>
<td><em>(Required)</em> This statement contains the utility control statements that you specify to tell the utility how to process exceptions and create solutions.</td>
</tr>
</tbody>
</table>
Coding control statements for the Solution utility

Control statements for the Solution utility contain commands and keywords with their associated values.

Specify these control statements in the SOLVE data set. See Figure 62 on page 343 for example control statements.

Figure 62: Solution utility control statements

```plaintext
//SOLVE DD *
USE RECON=IMSA.RECON1
DB +
  DBD=DI21* +
  ONLYNEW +
  NOWARN
RSRC +
  TYPE=CA +
  NAME=CAG*
RSRC +
  TYPE=LOG +
  NAME=IMSA
RSRC +
  TYPE=RCON
```

Syntax rules

The following syntax rules apply to control statements for the Solution utility:

- Specify each command on a separate line.
- Follow each command with the associated keywords and their values.
- Use the plus sign (+) to continue a set of commands and keywords to the next line.
- Specify an equal sign (=) between each keyword and its value.
- To specify a comment, specify an asterisk (*) in column 1.
- For some keywords, you can specify the keyword value as a fully qualified name or a mask. A mask can contain one or more percent signs (%) and asterisks (*). Use the percent sign to mask a single character. Use the asterisk to mask multiple characters. See the following examples:

<table>
<thead>
<tr>
<th>Mask</th>
<th>Mask description</th>
<th>Match examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Match any name.</td>
<td>A, ABC, ABCXYZ</td>
</tr>
<tr>
<td>ABC*</td>
<td>Match any name that begins with ABC.</td>
<td>ABC, ABCDEF, ABCXYZ</td>
</tr>
<tr>
<td>%</td>
<td>Match any nonblank, single-character name.</td>
<td>A, B, X, Z</td>
</tr>
<tr>
<td>Mask</td>
<td>Mask description</td>
<td>Match examples</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>ABC%%</td>
<td>Match any five-character name that begins with ABC.</td>
<td>ABCDE, ABC12</td>
</tr>
<tr>
<td>*XYZ</td>
<td>Match any name that ends with XYZ (including the name XYZ).</td>
<td>XYZ, ABCXYZ, XXYZ</td>
</tr>
<tr>
<td>%%XYZ</td>
<td>Match any five-character name that ends with XYZ.</td>
<td>ABXYZ, XXYZ</td>
</tr>
<tr>
<td><em>DEF</em></td>
<td>Match any name that contains DEF in any position in the name.</td>
<td>ABCDEF, DEFXYZ, DEF, ABDEFYZ</td>
</tr>
<tr>
<td>*DEF%</td>
<td>Match any name ending with DEF plus one more character.</td>
<td>DEFZ, ABCDEFZ, ADEFZ</td>
</tr>
<tr>
<td>ABC*XYZ</td>
<td>Match any name that begins with ABC and ends with XYZ.</td>
<td>ABCXYZ, ABCDEXYZ, ABCDXZY</td>
</tr>
</tbody>
</table>

### USE command and keyword

Use the required USE command to identify the IMSPLEX for which the utility should generate solutions.

For the USE command, specify the required RECON keyword as explained in Table 40 on page 344.

**Table 40: USE command keyword for the Solution utility**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Usage</th>
<th>Valid values</th>
<th>Default value</th>
<th>Keyword description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON</td>
<td>Required</td>
<td>RECON data set name (1 to 44 alphanumeric characters)</td>
<td>None</td>
<td>Identify the data set name of the RECON that is associated with the IMSPLEX.</td>
</tr>
</tbody>
</table>

### DB command and keywords

Use the DB command to specify that the utility should perform solution processing for databases.

You can specify multiple DB commands. For each DB command, you can specify optional keywords as explained in Table 41 on page 345. If you do not specify any keywords, the utility generates solutions for all database exceptions.
Table 41: DB command keywords for the Solution utility

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Usage</th>
<th>Valid values</th>
<th>Default value</th>
<th>Keyword description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREA</td>
<td>Optional</td>
<td>Name or mask</td>
<td>None</td>
<td>Limit solution processing to the specified Fast Path area. This keyword is valid only if you specify the DBD keyword.</td>
</tr>
<tr>
<td>DBD</td>
<td>Optional</td>
<td>Name or mask</td>
<td>None</td>
<td>Limit solution processing to the specified databases. If you specify this keyword, do not specify the DBGROUP keyword for the same DB command.</td>
</tr>
<tr>
<td>DBGROUP</td>
<td>Optional</td>
<td>Name</td>
<td>None</td>
<td>Limit solution processing to the databases in the specified group. If you specify this keyword, do not specify the DBD keyword for the same DB command.</td>
</tr>
<tr>
<td>LIST</td>
<td>Optional</td>
<td>None</td>
<td>None</td>
<td>List the generated solution statements in the SOLOUT data set in addition to the GENJCL data set.</td>
</tr>
<tr>
<td>NOWARN</td>
<td>Optional</td>
<td>None</td>
<td>None</td>
<td>Skip exceptions that have a warning (yellow) status.</td>
</tr>
<tr>
<td>ONLYNEW</td>
<td>Optional</td>
<td>None</td>
<td>None</td>
<td>Skip exceptions for which the BRIUBSOL utility has previously generated solutions.</td>
</tr>
</tbody>
</table>
RSRC command and keywords

Use the RSRC command to specify that the utility should perform solution processing for change accumulations, logs, and RECON data sets.

You can specify multiple RSRC commands. For each RSRC command, you can specify optional keywords as explained in Table 42 on page 346. If you do not specify any keywords, the utility generates solutions for all change accumulation, log, and RECON exceptions.

Table 42: RSRC command keywords for the Solution utility

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Usage</th>
<th>Valid values</th>
<th>Default value</th>
<th>Keyword description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PART</td>
<td>Optional</td>
<td>Name or mask</td>
<td>None</td>
<td>Limit solution processing to the specified HALDB partition. This keyword is valid only if you specify the DBD keyword.</td>
</tr>
</tbody>
</table>

Generating solution JCL on demand from the console

You can generate solution JCL on demand from the console.

When you perform this action for a selected RECON or database group, Advisor creates a job with two steps. The first step executes the Solution (BRIUBSOL) utility,
which creates GENJCL statements for the JCL Generation (CPCBATCH) utility, saves the statements to a temporary data set, and passes the data set to the second step. The second step executes the JCL Generation utility, which uses the saved GENJCL statements to generate solution JCL for all objects (database, change accumulation, log, and RECON data sets) that have one or more reported exceptions. The solution JCL is saved to a specified PDS. You can examine, edit, and submit the JCL, which corrects the reported exceptions.

To generate solution JCL on demand

1. In the navigation window, right-click a RECON or database group object.
2. From the right-click menu, choose Generate JCL.
3. In the Generate JCL window, select GENERATE SOLUTIONS FOR EXCEPTIONS.
4. Click Generate JCL.
5. In the popup window, specify the name of the existing partitioned data set to contain the generated JCL, and click OK.

The JCL for each object is generated and saved in the specified data set (as identified by the JCLPDS DD statement).

Interpreting return codes

The following table describes return codes that the Solution utility can generate.

<table>
<thead>
<tr>
<th>Return Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>The utility generated solutions for all objects that were specified in the DB and RSRC commands.</td>
</tr>
<tr>
<td>04</td>
<td>No exceptions were found for at least one DB or RSRC command.</td>
</tr>
<tr>
<td>08</td>
<td>The utility encountered an error condition.</td>
</tr>
<tr>
<td>14</td>
<td>The utility detected an error condition that was related to the RECON data sets.</td>
</tr>
<tr>
<td>16</td>
<td>One or more syntax errors were detected, a required DD statement was missing, or an abend condition occurred.</td>
</tr>
</tbody>
</table>
Using GENJCL output

You can use the generated GENJCL commands as input to the JCL Generation utility (program CPCBATCH).

Figure 63 on page 348 shows examples of GENJCL statements that are produced by the Solution utility.

Figure 63: GENJCL statements that are generated by the Solution utility

GENJCL RECON_DSN(BMCBRI.QA.BROA.ZEG$01.R91.RCON1 ) -
   SOLUTION(CHANGE ACCUM ) CLASS(BRI ) +
      FEATURE(CHANGE ACCUMULATION ) +
      MEMBER(CAGERR04) +
      POPUP01(CAGRNME=CAGERR04       ) +
   OUTFILE(JCLPDS)

GENJCL RECON_DSN(BMCBRI.QA.BROA.ZEG$01.R91.RCON1 ) -
   SOLUTION(DBRC COMMANDS ) CLASS(IBM ) +
      FEATURE(LIST LOG ) +
      MEMBER(IRME0001) +
      POPUP01(ITEMTIME=090710900       ) +
      POPUP02(ITEMDATE=2005094         ) +
   OUTFILE(JCLPDS)

GENJCL RECON_DSN(BMCBRI.QA.BROA.ZEF$02.R91.RCON1 ) -
   SOLUTION(DATABASE RECOVERY ) CLASS(BRI ) +
      FEATURE(RECOVER TO CURRENT ) +
      DBD(IRMIP2P ) PARTNAME(IP2P2 ) MEMBER(&DBD ) +
   OUTFILE(JCLPDS)

Figure 64 on page 348 shows the use of this output for an execution of the JCL Generation utility. For more information, see “Generating JCL with the JCL Generation utility” on page 350.

Figure 64: CPCBATCH job to generate JCL from BRIUBSOL-generated statements

```bash
//CPCBATCH: GENERATE JCL MEMBERS FOR ALL SOLUTIONS
//DELETE & ALLOCATE JCL PDS
//DELPDS EXEC PGM=IEFBR14
//JCLPDS DD DISP=(MOD,DELETE,DELETE),UNIT=SYSDA,
//      SPACE=(CYL,(1,1),RLSE),
//      DSN=your.JCLPDS
//ALLPDS EXEC PGM=IEFBR14
//JCLPDS DD DISP=(NEW,CATLG),SPACE=(CYL,(1,1,10),RLSE),
//      UNIT=SYSDA,
//      DCB=(DSORG=PO,RECFM=FB,LRECL=80),
//      DSNTYPE=LIBRARY,
//      DSN=your.JCLPDS
//BRIJCL07 EXEC PGM=CPCBATCH
//STEPLIB DD DISP=SHR,DSN=cpc.options.library
//      DD DISP=SHR,DSN=bmc.load.library
//JCLPDS DD DISP=SHR,DSN=your.JCLPDS
//CPCSYSSIN DD DISP=SHR,DSN=OUTPUT.FROM.BRIUBSOL
```
Using the SOLVE Processing Log report

The Solution utility produces the SOLVE Processing Log report and writes it to the SOLOUT data set.

This report lists all selected objects and their associated exceptions. When you specify the LIST control statement keyword, the report also lists the generated JCL statements.

Figure 65: SOLVE Processing Log report

```
//SOLVE PROCESSING LOG
START SOLUTION PROCESSING FOR: DBD=(all) 14:06:54.45
FIND SOLUTION(S) FOR:
   DBD   AREA/PART INDEX DSGID DD NAME EXCEPTION
   --------  --------  -----  --------    --------
   IRMIP3P   IP3P1   00001 IP3P1A      DBDS NOT IN ANY DBDS GROUP
   IRMIP3P   IP3P2   00001 IP3P2A      DBDS NOT IN ANY DBDS GROUP
   <data omitted>
   IRML87P   IRML87C0   IRML87C0    DBDS NOT IN ANY DBDS GROUP

190 Exception(s) for 46 Database object(s)
SOLUTION PROCESSING COMPLETED - 25 Solution Member(s) Generated 14:11:00.74
START SOLUTION PROCESSING FOR: TYPE=CA NAME=(any) 14:11:00.74
FIND SOLUTION(S) FOR:
   TYPE   NAME     DSNMNE     YYYY/MM/DD HH:MM:SS.THMIJU EXCEPTION
   ----  -------- ------                             ---------- ---------------  --------
   CA    CAGBRIH5                       MIN # OF CA RUNS NOT AVAILABLE
   CA    CAGBRIH6                       MIN # OF CA RUNS NOT AVAILABLE
   <data omitted>
SOLUTION PROCESSING COMPLETED - 13 Solution Member(s) Generated 14:12:10.07
START SOLUTION PROCESSING FOR: TYPE=LOG NAME=(any) 14:12:10.07
FIND SOLUTION(S) FOR:
   TYPE   NAME     DSNMNE     YYYY/MM/DD HH:MM:SS.THMIJU EXCEPTION
   ----  -------- ------                             ---------- ---------------  --------
   LOG   IRMEG$03 BMCIRM.QA.NOTCAT.BLOGP10 2005/04/04 09:06:55.800000 PRILOG ENTRY NOT CATALOGED
   LOG   IRMEG$03 BMCIRM.QA.NETCAT.BLOGS50 2005/04/04 09:07:16.400000 SECLOG ENTRY NOT CATALOGED
   <data omitted>
SOLUTION PROCESSING COMPLETED - 13 Solution Member(s) Generated 14:13:17.44

Replacing the DLIAJCLN utility with the BRIUBSOL utility

If you are using the DLIAJCLN utility to generate database solution control statements, you can continue to do so.

Although the DLIAJCLN utility now calls the BRIUBSOL utility to process the solutions, you do not need to change your JCL. However, you might want to convert to BRIUBSOL JCL to take advantage of additional flexibility and to process solutions for change accumulations, logs, and RECON data sets.
Generating JCL with the JCL Generation utility

You can use the JCL Generation utility (program CPCBATCH) to create JCL in batch. The JCL Generation utility creates JCL for an object (a database, partition, area, change accumulation group, or log) for a combination of solutions, classes, and features.

Information in the data repositories is used to construct the JCL, which can be executed immediately or can be saved in a sequential data set or a PDS.

The Solution utility can generate control statements for the JCL Generation utility. For more information, see “Handling exceptions with the Solution batch utility” on page 341.

Coding JCL for the JCL Generation utility

The following figure shows an example JCL for the JCL Generation utility.

**Figure 66: JCL Generation utility JCL**

```jcl
//JOBCARD JOB
//GENJCL EXEC PGM=CPCBATCH,REGION=4096
//STEPLIB DD DISP=SHR,DSN=cpc.options.library
// DD DISP=SHR,DSN=bmc.load.library
//JCLPDS DD DISP=SHR,DSN=bmc.genjcl.cntl
//CPCSYSIN DD *
GENJCL RECON_DSN(IMS.RECON1) -
  DBGROUP(ALL) DBD(CUSTOMER) -
  SOLUTION(STD REORG) CLASS(IBM) FEATURE(MULTI STEP DBIL REORG) -
  OUTFILE(JCLPDS)
/*
```

**EXEC and DD statements**

The following table lists statements that are required for execution of the JCL Generation utility.

**Table 44: JCL Generation utility JCL statements**

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td><em>(required)</em> You must specify PGM=CPCBATCH.</td>
</tr>
<tr>
<td>STEPLIB</td>
<td><em>(required)</em> In the STEPLIB concatenation, include the library that contains your Cross Product Connectivity (CPC) options module and the library that contains the BMC product load modules. APF authorization is not required.</td>
</tr>
</tbody>
</table>
Coding the GENJCL command and keywords

The command to execute the JCL Generation utility is `GENJCL`. Standard keywords and `POPUPxx` subkeywords control utility processing.

**Note**

Normally, you should use the Solution utility (program `BRIUBSOL`) to create the control statements for the JCL Generation utility. This utility quickly and reliably generates the appropriate combinations of GENJCL commands and keywords for selected objects.

### CPCSYSIN control statement syntax

The control statement data set must have the ddname `CPCSYSIN` and must contain 80-character fixed-length records.

A command can have keywords and comments, separated where necessary with separators and continuation characters (if statements continue to the next line). You can enter commands anywhere in positions 1 through 72 of the input statement (positions 73 through 80 are ignored).

### Commands

The control statement must begin with a valid command.

A separator must follow a command.

### Keywords

Keywords follow a command and invoke options.
All keywords are non-positional. You specify a keyword with a value following it in parentheses. The keyword value can be any character string up to 255 characters. All alphanumeric and special characters are allowed.

Comments

Comments consist of an alphanumeric character string beginning with a slash-asterisk (/*) and ending with an asterisk-slash (*/).

Comments cannot start in position 1 of an input statement.

Separators

When you require a separator, use a blank, a comma, or a comment.

More than one separator is allowed between keywords. Do not use a separator between a keyword and its value.

Continuation characters

Valid continuation characters are the plus sign (+) and the minus sign (-).

Use them to continue control statements and comments that do not fit on a single line of input. The continuation character must be the last nonblank character. Use the continuation characters as follows:

- Use the plus sign (with no spaces before it) to continue values for a single keyword to a second line (or subsequent lines). The plus sign deletes leading separators from the continued line.

- Use the minus sign (with a space before it) to continue a list of keywords for a single command. The minus sign does not delete leading separators from the continued line.

Standard keywords

The following table lists standard keywords for the JCL Generation utility.

---

**Note**

The JCL Generation utility requires that you specify a valid combination for SOLUTION, CLASS, and FEATURE keyword values. Table 37 on page 327 lists valid combinations.
### Table 45: JCL Generation utility keywords

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREANAME</td>
<td>(AREANAME) (ALL)</td>
<td>ALL</td>
<td><em>(optional)</em> Specify the name of the DEDB area. If specified, the DBD is required. If you omit AREANAME, all areas are processed.  For information about specifying a range of areas, see “POPUPxx subkeywords” on page 356.</td>
</tr>
<tr>
<td>CLASS</td>
<td>ADVISOR BRI CLASSIC EP IBM MIX OTHER</td>
<td>None</td>
<td>Specify the solution class. The solution, class, and feature must be a valid combination. For valid combinations, see “BMC-provided Advisor solutions” on page 326.</td>
</tr>
<tr>
<td>DBD or DBDNAME</td>
<td>(dbdname)</td>
<td>None</td>
<td>Specify the name of the database. If the DBGROUP keyword is specified, the DBD keyword is optional. If the PARTNAME or AREANAME keyword is specified, the DBD keyword is required. If the DBD keyword or the DBDNAME keyword is specified, JCL is generated for only the specified database. If DBGROUP is not specified, the owning database group is used.</td>
</tr>
<tr>
<td>DBGROUP</td>
<td>(Database group name)</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name of the database group. If the DBD keyword is specified, the DBGROUP keyword is optional. If DBD is not specified, JCL is generated for all databases in the database group for which the database type and the solution, class, and feature apply. If DBD is specified, JCL for the database in the DBGROUP is generated regardless of the owning database group.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FEATURE</td>
<td>(Solution feature)</td>
<td>None</td>
<td>Specify the solution feature. The solution, class, and feature must be a valid combination. For valid combinations, see “BMC-provided Advisor solutions” on page 326.</td>
</tr>
<tr>
<td>JOBCARD</td>
<td>S M</td>
<td>M</td>
<td><em>(conditional)</em> Specify whether to create a job statement for a single database or for multiple databases. The JOBCARD keyword applies only when the output data set is a sequential data set and JCL is being generated for multiple databases. To create a job statement for each database, specify M. Each database is a separate job. To create a single job statement for all databases, specify S. Each database will have one or more separate steps in the JCL.</td>
</tr>
<tr>
<td>MEMBER</td>
<td>(Output PDS member name)</td>
<td>DBD name, PART name, or AREA name MEMBER is required if the DBD, PART, or AREA name is not present for an output PDS or PDSE.</td>
<td>Specify the output PDS member name or name pattern. The MEMBER keyword controls the member names of the output data set when the data set is a PDS or PDSE. If the utility is generating JCL for multiple databases and the value of MEMBER is a specific name, the utility places JCL for all databases in that member. A value of ‘&amp;DBD’ indicates that utility should use the DBD name, partition name, or area name for the member name. The value can be a pattern for multiple PDS member names. If the last character of the value is an asterisk (<em>), the utility uses the preceding characters as a prefix; the characters from the asterisk through the eight characters of the name will be numeric values that begin with zeros. <strong>Example:</strong> MEMBER(DBD</em>) generates member names DBD00000, DBD00001, DBD00002, ...</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OUTDATASET or</td>
<td>(Output data set name)</td>
<td>None</td>
<td><em>(conditional)</em> Specify a name for the output data set. The OUTDATASET keyword is required if OUTFILE is not specified. OUTDATASET is mutually exclusive with OUTFILE. The value specifies the name of an existing data set in which to place the generated JCL. The data set is dynamically allocated during processing. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records. If the data set is partitioned, member names can be specified in parentheses following the data set name. The member name value can be any of the values described by the MEMBER keyword. <strong>Example:</strong> OUTDATASET(my.sol.jcl(&amp;DBD)) is equivalent to OUTDATASET(my.sol.jcl) MEMBER(&amp;DBD)</td>
</tr>
<tr>
<td>ODS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTFILE or OFILE</td>
<td>(Output data set ddname)</td>
<td>None</td>
<td><em>(conditional)</em> Specify the ddname for the output data set. The OUTFILE keyword is required if OUTDATASET is not specified. OUTFILE is mutually exclusive with OUTDATASET. The value specifies the ddname of a data set that is allocated in the JCL of the step which executes the GENJCL function. The generated JCL is placed in this data set. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records. The JCL statement can specify an internal reader for automatic submission of JCL.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Keywords and Descriptions

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUT_DSNAME</td>
<td>(Output data set name)</td>
<td>None</td>
<td><em>(conditional)</em> Specify a name for the output data set. The OUT_DSNAME keyword is required if the feature is Generate Solutions for Exceptions. The value specifies the name of an existing partitioned data set of 80-byte, fixed-length records. This data set is named in the JCLPDS DD statement of the output JCL.</td>
</tr>
<tr>
<td>PARTNAME</td>
<td>(Partition name)</td>
<td>None</td>
<td><em>(optional)</em> Specify the name of the database partition. If specified, the DBD keyword is required.</td>
</tr>
<tr>
<td>POPUPxx</td>
<td>(Subkeyword=value)</td>
<td>None</td>
<td>Specify the POPUPxx keyword with a subkeyword to provide additional values that control JCL Generation utility processing. For more information about coding the POPUPxx keyword and subkeywords, see “POPUPxx subkeywords” on page 356.</td>
</tr>
<tr>
<td>RECON_DSN</td>
<td>(RECON1 data set name)</td>
<td>None</td>
<td>Specify the RECON data set name for the IMS system that owns the database or database group.</td>
</tr>
<tr>
<td>RNAME</td>
<td>RECON</td>
<td>None</td>
<td>Specify the literal value RECON for resource solutions. Resource solutions include those for change accumulations, logs, and RECONs. The RNAME keyword is required when you do not specify the DBD keyword and DBGROUP keyword.</td>
</tr>
<tr>
<td>SOLUTION</td>
<td>(Solution name)</td>
<td>None</td>
<td>Specify the name of the solution. The solution, class, and feature must be a valid combination. For valid combinations, see “BMC-provided Advisor solutions” on page 326.</td>
</tr>
</tbody>
</table>

### POPUPxx subkeywords

The following table lists POPUPxx subkeywords that control JCL Generation utility processing.

Specify the POPUPxx keyword with a subkeyword and subkeyword value to pass the value to the generated JCL.
Note

If you specify more than one POPUPxx keyword in a control statement, you must use a different 2-digit suffix for each POPUPxx keyword that you specify. For example, if you specify POPUP01, you must choose from POPUP02 through POPUP10 for the next POPUPxx keyword that you specify.

Example:
POPUP01(subkeyword=xx)
POPUP02(subkeyword=xx)

Table 46: POPUPxx subkeywords for the JCL Generation utility

<table>
<thead>
<tr>
<th>POPUPxx Subkeyword</th>
<th>Accepted values for subkeyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREAREND=xx</td>
<td>2 to 2048</td>
<td>None</td>
<td>When specifying a range of Fast Path areas, specify the ending area number. AREAREND must be specified with AREARSTA. The AREAREND value that is specified must be greater than the AREARSTA value. <em>Example:</em> POPUP01(AREARSTA=10) POPUP02(AREAREND=20)</td>
</tr>
<tr>
<td>AREARSTA=xx</td>
<td>1 to 2047</td>
<td>None</td>
<td>When specifying a range of Fast Path areas, specify the beginning area number. AREARSTA must be specified with AREAREND. The AREARSTA value that is specified must be less than the AREAREND value. <em>Example:</em> POPUP01(AREARSTA=10) POPUP02(AREAREND=20)</td>
</tr>
<tr>
<td>CAGRPNAME=xxxxxxxx</td>
<td>1 to 8 alphanumeric characters</td>
<td>None</td>
<td>Specify a valid DBRC change accumulation (CA) group name. The following solution features use this subkeyword:  ■ CHANGE ACCUMULATION  ■ IC TRIGGERING BY IMAGE COPY  ■ IMAGE COPY BY CAGRP  ■ LIST CAGRP  ■ RECOVER TO CURRENT BY CAGRP  <em>Example:</em> POPUP01(CAGRPNAME=CAGRP01)</td>
</tr>
<tr>
<td>EXTDIOVF=xx</td>
<td>1 to 32765</td>
<td>0 (zero)</td>
<td>Specify the number of UOWs to add to the IOVF storage portion of the DEDB.  <em>Example:</em> POPUP01(EXTDIOVF=10)</td>
</tr>
<tr>
<td><strong>Subkeyword</strong></td>
<td><strong>Accepted values for subkeyword</strong></td>
<td><strong>Default value</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------</td>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>EXTDSDEP=xx</td>
<td>1 to 8388601</td>
<td>0 (zero)</td>
<td>Specify the number of CIs to add to the SDEP storage portion of the DEDB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Example:</strong> POPUP01(EXTDSDEP=100)</td>
</tr>
<tr>
<td>GRPNAME1=xx</td>
<td>1 to 8 alphanumeric characters</td>
<td>None</td>
<td>Specify a valid CA group or database data set (DBDS) group name. The following solution features use this subkeyword:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ CHANGE CAGRP ADD</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ CHANGE DBDSGRP ADD</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Example:</strong> POPUP01(GRPNAME1=DBGROUPA)</td>
</tr>
<tr>
<td>GRPNAME n=xxxxxxxx</td>
<td>1 to 8 alphanumeric characters</td>
<td>None</td>
<td>Specify a valid database data set (DBDS) group name. The CHANGE DBDSGRP REMOVE solution feature uses this subkeyword:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Example:</strong> POPUP01(GRPNAME2=DBGROUPB) POPUP02(GRPNAME3=DBGROUPC) POUP03(GRPNAME4=DBGROUPD) POPUP04(GRPNAME5=DBGROUPE)</td>
</tr>
<tr>
<td>ITEMDATE=yyy yddd</td>
<td>Julian date (4-digit year and 3-digit day of year)</td>
<td>None</td>
<td>Specify the Julian date of the item to be processed. You will often need to specify the ITEMDATE subkeyword and the ITEMTIME subkeyword together in the same control statement. The following solution features use this subkeyword:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ COPY IMAGE COPY</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ LIST LOG</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you are preparing JCL for the COPY IMAGE COPY solution feature, the date must exactly match the date in the timestamp of the image copy data set to be copied.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you are preparing JCL for the LIST LOG solution feature, the date is used as a starting (or &quot;from&quot;) date and does not need to match a log timestamp exactly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Example:</strong> POPUP01(ITEMDATE=2006055) POPUP02(ITEMTIME=0130000)</td>
</tr>
<tr>
<td>POPUP.xx Subkeyword</td>
<td>Accepted values for subkeyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| ITEMTIME=hh mmssst  | Timestamp (2-digit hour, 2-digit minute, 2-digit second, and 1-digit tenth of a second) | None | Specify the timestamp of the item to be processed. You will often need to specify the ITEMDATE subkeyword and the ITEMTIME subkeyword together in the same control statement. The following solution features use this subkeyword:  
  - COPY IMAGE COPY  
  - LIST LOG  
  If you are preparing JCL for the COPY IMAGE COPY solution feature, the time must exactly match the timestamp of the image copy data set to be copied. If you are preparing JCL for the LIST LOG solution feature, the time is used as a starting (or "from") time and does not need to match a log timestamp exactly.  
  **Example:** POPUP01(ITEMDATE=2006055) POPUP02(ITEMTIME=0130000) |
| PROCSDEP=xx         | LOGICAL PHYSICAL NONE         | None | Specify how SDEP segment data is to be processed.  
  **Example:** POPUP01(PROCSDEP=LOGICAL) |
| PROCSSET=xx         | YES NO                        | None | Specify whether subset pointers should be retained or cleared.  
  **Example:** POPUP01(PROCSSET=YES) |
| SUGGESTD=xxx        | 2 to 255 for GENMAX 2 to 1024 for GRPMAX 0 to 999 for RECOVPD | None | Specify a valid numeric value for the GENMAX, GRPMAX, or RECOVPD. The following solution features use this subkeyword:  
  - CHANGE CAGRP GRPMAX  
  - CHANGE DBDS GENMAX  
  - CHANGE DBDS RECOVPD  
  **Example:** POPUP01(SUGGESTD=045) |

**Generating JCL with the Utility Migration JCL Generator**

The Utility Migration JCL Generator (the JCL Generator) creates the JCL to run the basic functions of the following products. You can use the JCL Generator to get going quickly with the BMC database products for IMS.
<table>
<thead>
<tr>
<th><strong>Basic function</strong></th>
<th><strong>Product or products</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Make an image copy</td>
<td>IMAGE COPY PLUS</td>
</tr>
<tr>
<td>Recover a full-function database</td>
<td>RECOVERY PLUS for IMS</td>
</tr>
<tr>
<td>Perform a parallel unload for a full-function database</td>
<td>FAST REORG FACILITY/EP</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP Express for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS with Online/Defrag Feature</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/Online for IMS</td>
</tr>
<tr>
<td>Unload a Fast Path database</td>
<td>Fast Path Reorg/EP</td>
</tr>
<tr>
<td>Reload a full-function database</td>
<td>FAST REORG FACILITY/EP</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP Express for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS with Online/Defrag Feature</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/Online for IMS</td>
</tr>
<tr>
<td>Reload a Fast Path database</td>
<td>Fast Path Reorg/EP</td>
</tr>
<tr>
<td>Rebuild indexes for a full-function database</td>
<td>FAST REORG FACILITY/EP</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP Express for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS with Online/Defrag Feature</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/Online for IMS</td>
</tr>
<tr>
<td>Rebuild indexes for a Fast Path database</td>
<td>Fast Path Indexer/EP</td>
</tr>
<tr>
<td>Perform a single-step online reorganization</td>
<td>MAXM Reorg/Online for IMS</td>
</tr>
<tr>
<td>Perform a single-step offline reorganization</td>
<td>FAST REORG FACILITY/EP</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP Express for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/EP for IMS with Online/Defrag Feature</td>
</tr>
<tr>
<td></td>
<td>MAXM Reorg/Online for IMS</td>
</tr>
<tr>
<td>Perform an online reorganization of a Fast Path database</td>
<td>Fast Path Online Reorg/EP</td>
</tr>
<tr>
<td>Check pointers</td>
<td>POINTER CHECKER PLUS</td>
</tr>
<tr>
<td>Perform an offline analysis of a Fast Path database</td>
<td>Fast Path Analyzer/EP</td>
</tr>
<tr>
<td>Perform an online analysis of a Fast Path database</td>
<td>Fast Path Online Analyzer/EP</td>
</tr>
</tbody>
</table>

To use the JCL Generator, perform the following tasks:

1. “Starting the JCL Generator” on page 361
Starting the JCL Generator

The JCL Generator is a REXX EXEC residing in the product’s sample library. Use the following procedure to start the JCL Generator:

1. On the ISPF TSO Commands panel, execute the following command:

   \texttt{EX 'HLQ.prdSAMP(MXAMJCL0)'}

   The variable \textit{HLQ} is the high-level qualifier that was assigned when the BMC database products for IMS were installed. The \textit{prd} variable is the product code.

The Utility Migration JCL Generator main menu is displayed:

2. Continue to “Setting the JCL generation parameters” on page 361.

Setting the JCL generation parameters

Before selecting an option to generate JCL, predefine the JCL generation parameters.

The utility saves the parameters between sessions in the variable definition (VDEF) profile member. The utility then use these parameters to generate the JCL associated with each product function. If you do not predefine one or more parameters, you are prompted for the information during the generation process.
To set the JCL parameters

1 On the Utility Migration JCL Generator main menu, select the Setup JCL Generation parameters option.

The parameters menu is displayed:

```
Utility Migration JCL Generator
Command ===> __________________________________________________________________
Select a category below to set/change JCL generation parameters or press enter to go through everything sequentially.
_ 0. VDEF Profile Management (save/load ISPF variables)
   1. Database Selection
   2. IMS Environment Definition
   3. Job Card Parameters
   4. Database Advisor Script Libraries
   5. STEPLIB
   6. ACBLIB
   7. DBOLIB
   8. MDALIB
   9. PSBLIB
A. Work Files/Dynamic Allocation
B. Image Copy/Checkpoint Models
C. Output JCL Data Set
Show all panels  N
Y = Show all input panels when a utility is selected.
N = Skip panels if the data is already set.
```

2 Take one of the following actions:

- Select each option to predefine specific parameters.
- Press **Enter** to step through all of the options.

A panel is displayed for each option.

3 Provide the requested information (described in the following table) and press **Enter**.

---

362 *Database Products for IMS Advisors and Toolkit User Guide*
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VDEF Profile Management</td>
<td>Use this option to manage your variable definition (VDEF) profile. All input variables are stored in the VDEF profile. As part of JCL generation, you must specify an output JCL PDS on the Output JCL Data Set panel (ISPF variable MJOUTDSN). When JCL is generated, the REXX EXEC writes all ISPF input variables to a member named @BMCVDEF in the output JCL PDS.</td>
</tr>
<tr>
<td></td>
<td>■ Load/Save VDEF</td>
</tr>
<tr>
<td></td>
<td>You can save a copy of the VDEF elsewhere or load values from another VDEF. You can use the PDS field to specify the PDS that the product should read from or write to. If you have already specified an output JCL PDS, this field will be populated with that data set name.</td>
</tr>
<tr>
<td></td>
<td>If you specify the LOAD or L command on the Command line, the product searches the PDS for a member named @BMCVDEF. If the member is found, the utility reads the VDEF and sets all input variables to the values specified in the VDEF. Specifying the SAVE or S command on the Command line writes the current values of all input variables to @BMCVDEF.</td>
</tr>
<tr>
<td></td>
<td>■ Restore defaults</td>
</tr>
<tr>
<td></td>
<td>You can clear all predefined parameters by using the DEFAULTS command.</td>
</tr>
<tr>
<td></td>
<td>■ Initialize Input Values</td>
</tr>
<tr>
<td></td>
<td>You can also use this option to control VDEF automatic loads. By default, Import VDEF on startup is set to Y, which imports the parameters from the output JCL PDS member @BMCVDEF. Setting the field to N starts the JCL Generator with the default values.</td>
</tr>
<tr>
<td></td>
<td>If you want to load a different VDEF on startup, specify the name of the PDS that contains your VDEF in the PDS field under Alternate input VDEF. Instead of importing from the output JCL PDS, the utility will import the VDEF from this PDS.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Database Selection</td>
<td>Use this option to select which databases and partitions to include in the generated JCL.</td>
</tr>
<tr>
<td></td>
<td>On this panel, partition refers to either HALDB partitions or DEDB areas.</td>
</tr>
<tr>
<td></td>
<td>You can use ISPF masking to name databases and partitions. An * represents any number of characters, and % represents a single character.</td>
</tr>
<tr>
<td>Add</td>
<td>To add a mask, type the DBD mask on the DBD line and press Enter. If you want to select specific partitions or areas, you can also provide a PART mask</td>
</tr>
<tr>
<td></td>
<td>on the PART line. If no PART mask is given, an asterisk is automatically entered in the field. After pressing Enter, the mask is added to the list.</td>
</tr>
<tr>
<td>Delete</td>
<td>To delete a mask, type D in the Action field for the row containing the mask you wish to delete and press Enter.</td>
</tr>
<tr>
<td></td>
<td>If you want to have all generated JCL for areas or partitions under a DBD compiled into one member, set this field to Y. All JCL generated for</td>
</tr>
<tr>
<td></td>
<td>areas or partitions that belong to a DBD will be placed in a single member named after the DBD name.</td>
</tr>
<tr>
<td></td>
<td>For example, database DB001 is a HALDB with partitions PART1, PART2, and PART3. There will be a member in the output JCL PDS named DB001 that contains</td>
</tr>
<tr>
<td></td>
<td>the JCL for PART1, PART2, and PART3.</td>
</tr>
<tr>
<td>Include all areas/partitions</td>
<td>If you want to include the indexes in the JCL generation.</td>
</tr>
<tr>
<td></td>
<td>into output DBD JCL member</td>
</tr>
<tr>
<td>Include Index Databases</td>
<td>Set this field to Y to include the indexes in the JCL generation.</td>
</tr>
<tr>
<td>Include Logical Relations</td>
<td>Set this field to Y to include logically related databases in the JCL generation.</td>
</tr>
<tr>
<td>IMS Environment Definition</td>
<td>Use this option to specify your IMS RECONs, IMSID, RESLIB, and MODSTAT data sets.</td>
</tr>
<tr>
<td></td>
<td>If you have a second RESLIB to include, add it with the STEPLIB option. The RESLIB specified here is automatically added to the STEPLIB.</td>
</tr>
<tr>
<td>Job Card Parameters</td>
<td>Use this option to specify job card and other JCL parameters for the JCL generation job.</td>
</tr>
<tr>
<td></td>
<td>The information specified here is used for the preparation JCL as well as on the generated JCL.</td>
</tr>
<tr>
<td></td>
<td>You also need to specify the JCL SYSOUT class and the REGION values for the generated JCL.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Database Advisor Script Libraries | Use this option to specify the location of JCL Generator script libraries. If you have a library of custom script exits, you can specify it here and it will be included when the scripts are run.  
  ■ Add  
  To add a library to the list, type the data set name in the **Script Library** field and press **Enter**.  
  ■ Delete  
  To delete a library from the list, type **D** in the **Action** field for the row containing the library you wish to delete and press **Enter**.  
  ■ Order  
  Use MU and MD commands in the **Action** field to move a library up or down in the list. |
| STEPLIB              | Use this option to specify the STEP libraries.  
  ■ Add  
  To add a library to the list, type the data set name in the **Load Library** field and press **Enter**.  
  ■ Delete  
  To delete a library from the list, type **D** in the **Action** field for the row containing the library you wish to delete and press **Enter**.  
  ■ Order  
  Use MU and MD commands in the **Action** field to move a library up or down in the list. |
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACBLIB</td>
<td>Use this option to specify ACB libraries.</td>
</tr>
<tr>
<td></td>
<td>■ Add</td>
</tr>
<tr>
<td></td>
<td>To add a library to the list, type the data set name in the ACB Library field and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Delete</td>
</tr>
<tr>
<td></td>
<td>To delete a library from the list, type D in the Action field for the row containing the library you wish to delete and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Order</td>
</tr>
<tr>
<td></td>
<td>Use MU and MD commands in the Action field to move a library up or down in the list.</td>
</tr>
<tr>
<td>DBDLIB</td>
<td>Use this option to specify DBD libraries.</td>
</tr>
<tr>
<td></td>
<td>■ Add</td>
</tr>
<tr>
<td></td>
<td>To add a library to the list, type the data set name in the DBD Library field and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Delete</td>
</tr>
<tr>
<td></td>
<td>To delete a library from the list, type D in the Action field for the row containing the library you wish to delete and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Order</td>
</tr>
<tr>
<td></td>
<td>Use MU and MD commands in the Action field to move a library up or down in the list.</td>
</tr>
<tr>
<td>MDALIB</td>
<td>Use this option to specify MDA libraries.</td>
</tr>
<tr>
<td></td>
<td>■ Add</td>
</tr>
<tr>
<td></td>
<td>To add a library to the list, type the data set name in the MDA Library field and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Delete</td>
</tr>
<tr>
<td></td>
<td>To delete a library from the list, type D in the Action field for the row containing the library you wish to delete and press Enter.</td>
</tr>
<tr>
<td></td>
<td>■ Order</td>
</tr>
<tr>
<td></td>
<td>Use MU and MD commands in the Action field to move a library up or down in the list.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PSBLIB</td>
<td>Use this option to specify PSB libraries.</td>
</tr>
<tr>
<td></td>
<td>■ Add</td>
</tr>
<tr>
<td></td>
<td>To add a library to the list, type the data set name in the <strong>PSB Library</strong> field and press <strong>Enter</strong>.</td>
</tr>
<tr>
<td></td>
<td>■ Delete</td>
</tr>
<tr>
<td></td>
<td>To delete a library from the list, type <strong>D</strong> in the <strong>Action</strong> field for the row containing the library you wish to delete and press <strong>Enter</strong>.</td>
</tr>
<tr>
<td></td>
<td>■ Order</td>
</tr>
<tr>
<td></td>
<td>Use MU and MD commands in the <strong>Action</strong> field to move a library up or down in the list.</td>
</tr>
<tr>
<td>Work Files/Dynamic Allocation</td>
<td>Use this option to specify information used by the JCL Generator to allocate data sets.</td>
</tr>
<tr>
<td></td>
<td>■ High-level Qualifier</td>
</tr>
<tr>
<td></td>
<td>The high-level qualifier to use when a script needs to allocate a data set.</td>
</tr>
<tr>
<td></td>
<td>■ Use SMS to allocate</td>
</tr>
<tr>
<td></td>
<td>If you want to use SMS to allocate the data sets, specify <strong>Y</strong> in the <strong>Use SMS to allocate</strong> field and fill in one of the SMS class fields. If this field is set to <strong>Y</strong>, the <strong>Unit</strong> and <strong>VOLSER</strong> fields are optional. If you don't want to use SMS, specify <strong>N</strong> and use the <strong>Unit</strong> and <strong>VOLSER</strong> fields. The unit and VOLSER are required if SMS is set to <strong>N</strong>.</td>
</tr>
<tr>
<td></td>
<td>■ Sort and IDCAMS</td>
</tr>
<tr>
<td></td>
<td>The sort file unit name and space type as well as where your IDCAMS DELETE and DEFINE PDSs are located.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: The database you want to use must be defined to DBRC to generate data set names.</td>
</tr>
</tbody>
</table>
### Option Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Copy/Checkpoint Models</td>
<td>Use this option to specify dynamic allocation models.</td>
</tr>
<tr>
<td></td>
<td>■ IC1 Model</td>
</tr>
<tr>
<td></td>
<td>Define the IC1 model used by Image Copy Plus and Recovery Plus to allocate the Image Copy output data sets. An IC1 model is required.</td>
</tr>
<tr>
<td></td>
<td>■ IC2 Model</td>
</tr>
<tr>
<td></td>
<td>The model to use if the ICDUAL or RVPDUAL option (found on the Image Copy/Recovery specific options panels) is set to Y.</td>
</tr>
<tr>
<td></td>
<td>■ Checkpoint Model</td>
</tr>
<tr>
<td></td>
<td>The model to use for the checkpoint data set when using Automatic Restart.</td>
</tr>
<tr>
<td></td>
<td>To edit a model, type S in the selection field and press Enter.</td>
</tr>
<tr>
<td></td>
<td>Panels are displayed on which to specify the dynamic allocation options that are a subset of the options available for an Image Copy Plus Dynamic Allocation Model.</td>
</tr>
<tr>
<td></td>
<td>For more information about the options, see the Backup and Recovery Products for IMS Image Copy Utility User Guide. Only the first panel contains required parameters.</td>
</tr>
<tr>
<td></td>
<td>Subsequent panels contain optional parameters.</td>
</tr>
<tr>
<td></td>
<td>On the first panel, specify either a data set name or data set prefix and an allocation method.</td>
</tr>
<tr>
<td>Output JCL Data Set</td>
<td>Use this option to specify the name of the output JCL PDS.</td>
</tr>
<tr>
<td></td>
<td>The output JCL PDS is where you want the generated JCL and VDEF to be written. If the specified PDS does not exist, fill in the allocation options. If the PDS exists, set the Overwrite existing PDS field to Y to write to the same PDS.</td>
</tr>
</tbody>
</table>

4 In the Show all panels field, specify if you want all predefined panels to be displayed as you generate JCL:

- **N (default)**
  
  During the JCL generation process, do not display a relevant panel if all the required parameters were predefined. If parameters were not predefined, the relevant panel will be displayed.

- **Y**
  
  During the JCL generation process, display all relevant panels. Any parameters that you predefined will be displayed on each panel.

5 Press Enter.

You are asked whether to save the predefined parameters.

6 Continue to “Generating JCL” on page 369.
Generating JCL

Use the following procedure to generate JCL:

1. On the Utility Migration JCL Generator main menu, select the option corresponding to the function for which you want to generate JCL:

<table>
<thead>
<tr>
<th>Option</th>
<th>Creates JCL to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Copy Plus</td>
<td>Copy a database image</td>
</tr>
<tr>
<td>Recovery Plus</td>
<td>Recover a database</td>
</tr>
<tr>
<td>Parallel Unload / Fast Path Offline Unload</td>
<td>Unload a parallel database</td>
</tr>
<tr>
<td>Reload / Fast Path Offline Reload</td>
<td>Reload a database</td>
</tr>
<tr>
<td>Rebuild Indexes</td>
<td>Rebuild database indexes</td>
</tr>
<tr>
<td>Single Step Online REORG</td>
<td>Reorganize an online database in a single step</td>
</tr>
<tr>
<td>Single Step Offline REORG</td>
<td>Reorganize an offline database in a single step</td>
</tr>
<tr>
<td>Fast Path Online REORG</td>
<td>Reorganize a database</td>
</tr>
<tr>
<td>Pointer Check / Fast Path Analyze Offline Database</td>
<td>Check the pointers and analysis an offline database</td>
</tr>
<tr>
<td>Fast Path Analyze Online Database</td>
<td>Analyze a database</td>
</tr>
</tbody>
</table>

2. On the displayed panel, specify the required information, and press 

   **Enter**.

   *Note*

   The following options do not display a function-specific panel:
   - Rebuild Indexes
   - Single Step Online REORG
   - Single Step Offline REORG

   If *show all panels* was specified on the JCL Generation Parameters menu, additional panels are displayed.

3. On the Output JCL Data Set panel, verify the output JCL information and press 

   **Enter**.

   The JCL is generated into the specified data set.

   If you specify a large number of databases, areas, or partitions, or you specify masks that match a large number of databases, areas, or partitions, be sure to specify an output PDS with the appropriate space and directory blocks.
For extra customization, you can use the JCL script language to modify or add lines to the generated JCL. The JCL script language is documented in the *Database Products for IMS Advisors and Toolkit User Guide*.

## Generating and working with recovery JCL

The Recovery Wizard provides a step-by-step process for recovering a group of databases. The Workflow Status repository stores Recovery Wizard requests. The Workflow Status window displays these requests and provides access to the generated JCL.

For more information, view the Quick Course "Recovery Advisor for IMS - Using the JCL Generation Wizard."

### Recovery Wizard

You can use the Recovery Wizard to create JCL for executing the BMC Recovery utility for all databases that belong to a database group.

You choose a group to recover, the recovery point, and whether to create JCL that simulates a recovery or performs an actual recovery. In addition, you can choose to recover to alternate data sets. You can accept established values for recovery parameters or specify override values. You must also specify an output data set and member to contain the generated JCL. After you have provided the requested information, the wizard generates the JCL in the background, writes the JCL to the specified output data set and member, and notifies you when the process is complete.

To use the Recovery Wizard, you must have a license for Backup and Recovery Solution for IMS.

### Workflow Status repository

The Recovery Wizard writes status entries to the Workflow Status repository.

A status entry contains information about the Recovery Wizard request and provides access to the generated JCL in the output data set. The entries are self-maintaining; after a specified number of days, the old entries are removed as new entries are created. You can control the number of days by using the Detail Record...
Retention (days) parameter in the Repository Management branch of the Maintain Parameters window.

Workflow Status window

You can use the Workflow Status window to work with previously submitted Recovery Wizard requests.

You can access the Workflow Status window by clicking the link in the notification popup dialog or by right-clicking IMS Databases in the Navigation window. If you want to limit the view to the actions that were executed for one RECON, you can right-click that RECON object and select Workflow Status.

The window obtains information about Recovery Wizard requests from status entries in the Workflow Status repository. Depending on your level of access to the output data set containing the generated JCL, you can view the status entry for the request and view, edit, and submit the generated JCL:

- If you have read and write access to the output data set, you can view the entry in the window, edit the JCL, and submit the JCL.
- If you have read-only access to the output data set, you can view the entry, browse the JCL, and submit the JCL.
- If you have no access to the output data set, you cannot view the entry, browse or edit the JCL, or submit the JCL.

**Note**

If you want a user (such as an operator) to be able to browse and submit JCL but not change it, grant this user read access to the output data set.

If you want to segregate different groups of users, so that each group can view only the requests for databases that belong to that group, you can create a different output data set for each group.

Generating recovery JCL

Use the following procedure to generate recovery JCL. For complete information about the Recovery Wizard, see the online Help.

1. In the Navigation window, navigate to the database group object that you want to recover.

2. Right-click the database group, and select **Recovery Wizard**.
3 Provide the requested information on each page of the wizard.

4 On the last page, click **Next** to start the JCL generation process.

When the process is complete, the wizard displays a notification dialog. The results of your request are available in the Workflow Status window.

**Working with recovery requests**

The following procedure is an overview. For complete information about the JCL Generation Status window, see the online Help.

1 Access the Workflow Status window with either of the following actions:
   - Click the link in the notification dialog from the Recovery Wizard.
   - In the Navigation window, right-click the IMS Databases object and select **Workflow Status**.

2 Click the status entry that corresponds to your request.

   **Note**

   The window highlights in bold the requests that you have not yet viewed. When you log on to the console, the console attempts to notify you of unviewed requests until you mark them as viewed. When you click an unviewed request, the console marks the request as viewed and no longer attempt to notify you about this request.

3 If the request has a CREATED status, depending on your level of access, you can click the **Edit JCL** or **Browse** button to edit, view, or submit the JCL.

4 To display jobs that you have submitted, click the **JES Viewer** button.

**Viewing jobs in the console**

The Jobs perspective and the JES viewer provide the convenience of viewing the JES log through the console. The Jobs perspective and the JES viewer provide direct access to the JES log. You can use the Jobs perspective and the JES viewer to look at the status of current jobs and to review the output for jobs that have completed. By default, Jobs perspective and the JES viewer list all jobs in the JES log that were submitted from your TSO user ID in order by job name. You can filter the list of jobs by full job name or partial job name.
Jobs in the Jobs perspective and the JES viewer are not limited to IMS jobs or jobs that are submitted through the console. You can view information about all jobs that write output to the JES log if you have authorization to view those jobs.

**Viewing jobs and refreshing job statuses**

You can either open the Jobs perspective or open the JES viewer in the Navigation window.

1. Open the Jobs perspective or the JES viewer.
   - To open the Jobs perspective, in the main navigation bar, click the **Jobs** tab.
     - If you have not already created a job filter, you need to create a job filter (see Filtering jobs that are displayed on page 374).
     - The Jobs perspective (Figure 67 on page 373) is displayed.
You can open the JES viewer from various places in the console. For example, you can click **JES Viewer** in the Workflow Status window.

The JES Viewer window (Figure 68 on page 374) is displayed. Because the default is to list all jobs that were submitted from your TSO user ID, you might have to wait a few seconds for the JES viewer to be populated.

**Figure 68: JES Viewer window**

2 To display the most current information that is logged to JES by refreshing information that is displayed, in the Jobs perspective in the JES viewer, click ✕. In the JES viewer, click **Refresh Job Status**.

### Filtering jobs that are displayed

Filtering jobs that are displayed in the Jobs perspective or the JES viewer is a good way to control the length of the list and the amount of time that is required for refreshing the list.

**To manage filters for jobs that are displayed in the Jobs perspective**

1 Add or edit a filter.

   - To add a filter, click the **Add Job Filter** icon.

   - To edit a filter, select a filter and click the **Edit Job Filter** icon.
The Add Job Filter or the Edit Job Filter dialog box prompts you for filter criteria.

**Figure 69: Add Job Filter dialog box**

The Edit Job Filter dialog box is similar to the Add Job Filter dialog box. In the Add Job Filter dialog box, you can select a host, and you can add a connection by clicking the **Add connection** icon. In the Edit Job Filter dialog box, the **Host** field and the **Add connection** icon are disabled. You cannot change the host for an existing filter and you cannot add a connection. To filter jobs for a different host, you need to add a filter.

1. In the Add Job Filter or Edit Job Filter dialog box, specify a job name or a job name mask and click **OK**.

If you have added a filter, the new filter appears in the Jobs perspective and jobs that meet the specified criteria are displayed. If you have edited a filter, jobs that meet the new criteria are displayed.

**To filter jobs that are displayed in the JES viewer**

1. In the JES Viewer window, click **Change Job Selection Criteria**.

The JES Viewer Selection dialog box prompts you for selection criteria.

**Figure 70: JES Viewer Selection dialog box**

2. In the JES Viewer Selection dialog box, specify a job name or a job name mask and click **OK**.

Jobs that meet the specified criteria are displayed in the JES viewer.
Viewing job output

Viewing job output is a good way to verify the result of a job and to research unexpected results.

**To view job output in the Jobs perspective**

1. In the jobs window of the Jobs perspective, select a job and click the View Output Data Sets icon.

   The output data set list is displayed (Figure 71 on page 376).

**Figure 71: Output data set list**

<table>
<thead>
<tr>
<th>ID</th>
<th>DD Name</th>
<th>Records</th>
<th>Data Set Name</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>JESMS6LG</td>
<td>20</td>
<td>RDAMPR.$103DWNLJOB...</td>
<td>03/21/2016 10:58:10</td>
</tr>
<tr>
<td>3</td>
<td>JESXCL</td>
<td>88</td>
<td>RDAMPR.$103DWNLJOB...</td>
<td>03/21/2016 10:58:10</td>
</tr>
<tr>
<td>4</td>
<td>JESYSMSG</td>
<td>245</td>
<td>RDAMPR.$103DWNLJOB...</td>
<td>03/21/2016 10:58:10</td>
</tr>
<tr>
<td>103</td>
<td>SYSPRINT</td>
<td>183</td>
<td>RDAMPR.$103DWNLJOB...</td>
<td>03/21/2016 10:58:10</td>
</tr>
<tr>
<td>106</td>
<td>OUTPUT</td>
<td>365</td>
<td>RDAMPR.$103DWNLJOB...</td>
<td>03/21/2016 10:58:10</td>
</tr>
</tbody>
</table>

2. Select a job ID and click the Browse Job Output icon. You can also double-click the job and data set.

   A job output window (Figure 72 on page 377) opens and shows the job output.
Each time that you select a different job and click the **Browse Job Output** icon, a new job output window opens.

**Figure 72: Job output window (Jobs perspective)**

---

**Tip**

To return from the list of output data sets to the jobs list, click the **Return to Jobs List** icon.

---

**To view job output in the JES viewer**

1. In the JES Viewer window, select a job with status OUTQ and click **Browse**.
When the Job Output window is first displayed (Figure 73 on page 378), only one tab (Data Set Selection) is displayed. The Data Set Selection tab lists all ddnames that contain output.

**Figure 73: Job Output window (initial display, JES viewer)**

<table>
<thead>
<tr>
<th>ID</th>
<th>DD Name</th>
<th>Step Name</th>
<th>Records</th>
<th>Data Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>CPCLOG</td>
<td>BRI4ADV</td>
<td>12</td>
<td>STCUSER.BRI4PROC.STC2...</td>
</tr>
<tr>
<td>103</td>
<td>CPCLOG</td>
<td>BRI4ADV</td>
<td>124</td>
<td>STCUSER.BRI4PROC.STC2...</td>
</tr>
<tr>
<td>101</td>
<td>PRTJCLXX</td>
<td>BRI4ADV</td>
<td>59</td>
<td>STCUSER.BRI4PROC.STC2...</td>
</tr>
</tbody>
</table>

2. On the Data Set Selection tab, select a ddname in the DD Name table column and click **View Output**.

A tab corresponding to the ddname is added to the Job Output window (Figure 74 on page 378) and shows the job output for that ddname.

Each time that you select a different ddname on the Data Set Selection tab and click **View Output**, a tab is added to the Job Output window (Figure 74 on page 378).

**Figure 74: Job Output window (JES viewer)**
Tip
You can navigate among tabs by clicking the tab that corresponds to the ddname for which you want to view data.
To close a tab, click the X control on the tab label.

Maintaining solutions

You can view information about BMC solutions and user-defined solutions. You can add and delete user-defined solutions.

Viewing solution information

You can view information about the solutions that are available in your environment. Solution definitions are used by the solution process to determine whether a solution is appropriate for a problem.

1 In the Navigation window, right-click the host object and choose Configuration => Maintain Solutions.
The Maintain Solutions window (Figure 75 on page 380) is displayed.

**Figure 75: Maintain Solutions window**

<table>
<thead>
<tr>
<th>Solution</th>
<th>Feature</th>
<th>Class</th>
<th>Type</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYZER</td>
<td>OFFLINE DB</td>
<td>FP</td>
<td>BMC</td>
<td>BFPAKAD</td>
</tr>
<tr>
<td>ANALYZER</td>
<td>OFFLINE IC</td>
<td>FP</td>
<td>BMC</td>
<td>BFPAKAI</td>
</tr>
<tr>
<td>ANALYZER</td>
<td>ONLINE</td>
<td>FP</td>
<td>BMC</td>
<td>BFPAKAN</td>
</tr>
<tr>
<td>BATCH CMD INTERFACE</td>
<td>START DATABASE</td>
<td>BRI</td>
<td>BMC</td>
<td>BRSCMD</td>
</tr>
<tr>
<td>BUILD</td>
<td>OFFLINE</td>
<td>FP</td>
<td>BMC</td>
<td>BFPELDO</td>
</tr>
<tr>
<td>CHANGE</td>
<td>OFFLINE IC</td>
<td>FP</td>
<td>BMC</td>
<td>BFPCGRI</td>
</tr>
<tr>
<td>CHANGE ACCUM</td>
<td>CHANGE ACCUM WITH DBRC STEP</td>
<td>BRI</td>
<td>BMC</td>
<td>BRSCADBR</td>
</tr>
<tr>
<td>CHANGE ACCUM</td>
<td>CHANGE ACCUMULATION</td>
<td>BRI</td>
<td>BMC</td>
<td>BRSCA</td>
</tr>
<tr>
<td>DATABASE RECOVERY</td>
<td>RECOVER TO CURRENT</td>
<td>BRI</td>
<td>BMC</td>
<td>BRSCREC</td>
</tr>
<tr>
<td>DATABASE RECOVERY</td>
<td>RECOVER TO CURRENT</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSCIBM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE CAGRP ADD</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSCAGA</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE CAGRP GRPMAX</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSCAGRM</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDS GENMAX</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSDLDCGN</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDS RECOVPD</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSCHRCGP</td>
</tr>
<tr>
<td>DBRC COMMANDS</td>
<td>CHANGE DBDSGRP ADD</td>
<td>IBM</td>
<td>BMC</td>
<td>BRSDLDSGA</td>
</tr>
</tbody>
</table>

Solution definitions are listed for each solution.

2. Select the table row of the solution for which you want to view information, and click **Browse**.
The Browse Solution Definition dialog box (Figure 76 on page 381) lists the solution definition for the selected solution.

**Figure 76: Browse Solution Definition dialog box**

---

**Adding a user solution**

To add a user solution, perform the following steps in the Maintain Solutions window.

1. In the Navigation window, right-click the host connection object and choose **Configuration => Maintain Solutions**.

2. In the Maintain Solutions window, click **Add**.
The Add Solution Definition dialog box (Figure 77 on page 382) is displayed. Use this dialog box to specify a script and to define characteristics of a user solution.

**Figure 77: Add Solution Definition dialog box**

![Add Solution Definition dialog box](image)

The dialog box has the following tabs:

- The Solution tab lists parameters that define support for full-function databases.
- The HALDB tab lists parameters that define support for HALDBs.
- The Popup Variables tab lists variables for the solution.

**Note**

If you select Yes for the **Database Group JCL Generation** field, the script generates JCL for all databases in the group. The JCL output file (in batch) must be a sequential file instead of a PDS.

3 Specify a solution definition, and click **Add**.
Deleting a user solution

You can delete a user solution in the Navigation window

1. In the Navigation window, right-click the host connection object and choose Configuration => Maintain Solutions.

2. In the Maintain Solutions window, click Delete.
   
The Delete Solution Definition dialog box is displayed.

3. Click Delete.
   
The user solution is deleted.
Controlling conditional processes

This section contains information about using the IC Triggering by CA feature and the Conditional Image Copy feature of Backup and Recovery Solution for IMS and the Conditional Reorg feature of Database Advisor.

Using the IC Triggering by CA feature

You can use the IC Triggering by CA feature to balance change accumulation (CA) and image copy (IC) processing for one or more CA groups.

The feature manages the size of the CA data set for a CA group through automated IC processing of the objects in the group. This automation ensures that the CA data set does not grow too large to be processed effectively during database recovery. See Figure 78 on page 386.
Note
This feature requires a license for the Backup and Recovery Solution for IMS product.

Figure 78: Flow of the IC Triggering by CA feature

Start BMC Change Accum utility

Have current statistics exceeded thresholds?

Yes

Start BMC Image Copy utility

Is the object flagged for image copy?

Yes

No

In repository, clear image copy flag from objects

In repository, flag objects for image copy, and post exceptions to exception list

No

Has it been too long since the last image copy?

Yes

Create image copy

No

Bypass image copy
Recommended usage of the IC Triggering by CA feature

To use the IC Triggering by CA feature effectively, set up matching BMC Change Accumulation utility and BMC Image Copy utility jobs to process CA groups on a regular basis. Enable the feature in each matching job.

For example, if you run the Change Accumulation utility periodically during the day, run the Image Copy utility to copy the same CA groups at the end of the day. With this feature enabled, the end-of-day Image Copy job usually does not create any image copies because the size of the CA data set normally does not exceed the established threshold. When the CA data set size exceeds the threshold, the Image Copy utility copies selected objects according to the options that you set for the feature.

After an image copy of an object, change records for that object are no longer needed for recovery because the changes are present in the image copy. The next change accumulation does not include those records, and the CA data set becomes smaller by a corresponding amount.

Implementing the IC Triggering by CA feature

To implement the IC Triggering by CA feature, set IC Triggering by CA parameters in the console and set the SMARTCA keyword for the Change Accumulation utility and Image Copy utility.

To set IC Triggering by CA parameters in the console

1. In the Navigation window, right-click the object (host, IMSPLEX, or CA group) for which you want to manage parameters.
2. Choose Configuration => Maintain Parameters.
3. In the tree at the left side of the Maintain Parameters window, select IC Triggering by CA.
4. Set parameter values as follows:
   a. To enable the IC Triggering by CA feature, set the Analyze CA Group to Trigger ICs parameter to Yes.
   b. To establish threshold values for the feature, set the CA File Size Threshold parameter. When the Change Accumulation utility processes a CA group, the utility compares the uncompressed size of the CA data set to the specified threshold value. An exception occurs if the size exceeds the threshold value.
To specify the action to take if the threshold is exceeded, set the **When CA file size threshold is exceeded** parameter. The Change Accumulation utility uses the specified technique to select eligible objects in the CA group for subsequent image copying when the Image Copy utility processes the CA group.

d To control whether the Image Copy utility copies any object in the CA group if the specified number of days has passed since the last image copy of that object, set the **Select any member of CA group for IC if "nn" days since last IC** parameter.

e To control whether the Image Copy utility copies all objects in the CA group that belong to the same full-function database or HALDB partition, set the **Select members of the CA group by DBD or Partition** parameter.

5 Close the Maintain Parameters window, and click **Yes** to save your changes.

**To set the SMARTCA keyword for the Change Accumulation utility**

1 Perform either of the following actions:
   - In a global options module that applies to the Change Accumulation utility job step, enter **Y** in the **IC Triggering by CA** field.
   - In the CAPSYSIN control statement data set, specify the SMARTCA(Y) keyword.

**To set the SMARTCA keyword for the Image Copy utility**

1 Perform either of the following actions:
   - In a global options module that applies to the Image Copy utility job step, enter **Y** in the **IC Triggering by CA** field.
   - In the ICPSYSIN control statement data set, specify the SMARTCA(Y) keyword.

**Change Accumulation utility processing**

When the IC Triggering by CA feature is enabled for change accumulation groups within a utility job step, the Change Accumulation utility performs the following types of processing:

- Collects, analyzes, and records change accumulation statistics for data set groups
- Compares these statistics to established thresholds
For objects that exceed thresholds, creates entries in Advisor repositories that flag these objects as requiring image copies and posts exceptions to the Exception List.

The Database Advisor infrastructure (repositories and servers) must be accessible during execution of the Change Accumulation utility job step, and a password for the Backup and Recovery Solution must be installed and active. If the infrastructure or password is not available, the utility issues a warning message, and change accumulation processing continues without the IC Triggering by CA feature.

**Image Copy utility processing**

When the IC Triggering by CA feature is enabled and the members of the change accumulation groups that were previously processed by the Change Accumulation utility are specified for image copy, the Image Copy utility selects members only if all of the following conditions are true (all other members in the change accumulation groups are bypassed):

- The Image Copy utility can verify the IC Triggering by CA status of the object through the Database Advisor infrastructure (repositories and servers). The infrastructure must be accessible during execution of the job step, and a password for the Backup and Recovery Solution must be installed and active.
- The image copy data set is allocated dynamically in the Image Copy utility job step.
- The job step requests a batch or online image copy (by specifying the IC, AIC, OIC, or AOIC function), and one of the following conditions is true:
  - The Change Accumulation utility has flagged the object as needing an image copy, based on parameters values that you set for the feature.
  - The age of the latest image copy is greater than or equal to the value that is specified with the Select any member of CA group for IC if "nn" days since last IC parameter in the console.

The utility performs an image copy of all data set groups in a DBD or partition if both of the following conditions exist:

- The Select members of the CA group by DBD or Partition parameter is set to Yes in the console.
- At least one data set group is eligible to be copied.

If you have specified a subsequent Copy Image Copy (CIC) function to use the image copy data set from an image copy function that the utility has bypassed, the utility also bypasses the CIC function.
Exceptions and solutions related to the IC Triggering by CA feature

If the Change Accumulation utility detects that the CA File Size Threshold value has been exceeded for a CA group, the utility reports the IC RECOMMENDED BY CHANGE ACCUM exception to the Exception List.

When the Image Copy utility takes an image copy of any object in the CA group, the utility clears the exception.

The Solutions window can recommend the IC TRIGGERING IMAGE COPY solution to correct the IC RECOMMENDED BY CHANGE ACCUM exception. You can also select this solution from the Generate JCL window. The solution generates JCL for executing the Image Copy utility to process the CA group with the SMARTCA(Y) keyword.

BMC recommends that if you run the Change Accumulation utility periodically (for example, during the day), you match that run with an Image Copy run (for example, at night) for the same CA groups and use the SMARTCA(Y) keyword. With the matched runs, the Image Copy run usually does not take any image copies. However, when the IC RECOMMENDED BY CHANGE ACCUM exception is detected, the Image Copy run takes the appropriate image copies.

Using the Conditional Image Copy feature

The Conditional Image Copy feature allows the BMC Image Copy utility (which is available in the Backup and Recovery Solution and the IMAGE COPY PLUS product) to bypass the performance of a scheduled batch image copy task for an object (a database, partition, or area).
See Figure 79 on page 391.

**Figure 79: Flow of the Conditional Image Copy feature**

The task is bypassed if both of the following conditions exist:

- The object has not been updated since the last image copy was performed.
- The last image copy is not too old.

The purpose of the feature is to avoid taking unnecessary and extraneous image copies, saving processing and data storage resources with no compromise to data integrity.
The Conditional Image Copy Simulation utility can help you estimate the extent to which the Conditional Image Copy feature can benefit your organization. For more information, see “Using the Conditional Image Copy Simulation utility” on page 395.

Conditional image copy processing

When an Image Copy utility job step is executed, the utility determines whether the Conditional Image Copy feature is active for each batch image copy task that is requested in the utility control statements.

Activation of the feature is controlled by the Skip IC if no updates parameter in the Maintain Parameters window and the SMARTIC keyword in the utility control statements (or global options module).

**Note**

The value of the SMARTIC keyword enables or disables the Conditional Image Copy feature and must be used in combination with the Skip IC if no updates parameter in the Maintain Parameters window. When the Conditional Image Copy feature is enabled in the utility, the Skip IC if no updates parameter selects the specific object (partition, or area, database) or group of objects (database group, IMSPLEX, host) for which conditional image copy processing will be performed. When the Conditional Image Copy feature is disabled in the utility, all other parameters are ignored and no conditional image copy processing occurs.

If the feature is active, the feature examines records in the RECON data sets to determine whether an image copy is required for the object. An image copy is required if any of the following conditions is true:

- The object has been allocated for update since the most recent image copy was taken (a database allocation record has a later time stamp than the most recent image copy record).
- The ICNEEDED flag is on for the object.
- The most recent record for the object is a NOTIFY.RECOV record.
- The number of days that have elapsed since the most recent image copy of the object is greater than or equal to the value of the Perform IC if "nn" days since last IC parameter.
Note
If a database is updated infrequently, this parameter can help ensure that an image copy is taken often enough to ensure that you meet other requirements that are related to backup frequency. For example, your disaster recovery plan might require a full backup of all databases to be performed at least once a month.

If the examination of the RECON data sets indicates that an image copy is required, the utility proceeds with the image copy task. If the image copy is not required, the utility bypasses the task.

Conditional image copy considerations

The following conditions apply to the use of the Conditional Image Copy feature:

- A license for Backup and Recovery Solution for IMS is required (a BRI-type password must be installed).

- The object must be registered to DBRC and registered in the BMC Database Management console.

- The Advisor environment (including the CPC, UIM, and ADV servers) must be active during execution of the Image Copy utility. If the environment is not active, the image copy is performed as scheduled.

- The Conditional Image Copy feature applies only to stand-alone image copy functions that are performed by the BMC Image Copy utility:
  - The IC or AIC command (in the control statements for the Image Copy utility) must be used to request the batch image copy function. (IIC, AIIC, OIC, AOIC, CIC, and ACIC commands are not supported.)
  - The feature does not apply to image copies that are invoked during execution of other functions (such as database recovery or reorganization).
  - The IBM IMS Database Image Copy utility (program DFSUDMP0) is not supported.

- If the SMARTIC keyword is specified in the control statements (or the global options module) for the Image Copy utility, the keyword value must be Y. If the value is N, the Conditional Image Copy feature is inactive for the task. The value of the SMARTIC keyword overrides the value of the Skip IC if no updates parameter.

- When an image copy function is bypassed, all functions that are associated with that function are also bypassed. For example, pointer checking is not performed.
Implementing the Conditional Image Copy feature

You can implement the Conditional Image Copy feature using Conditional Image Copy parameters in the console and setting the SMARTIC keyword for the Image Copy utility.

To set Conditional Image Copy parameters in the console

1. In the Navigation window, right-click the object (host, IMSPLEX, database group, database, partition, or area) for which you want to manage parameters.

2. Choose Configuration => Maintain Parameters.

3. In the tree at the left side of the Maintain Parameters window, select the Conditional IC branch.

4. Set values for the Conditional Image Copy parameters as follows:
   - To activate the Conditional Image Copy feature, set the Skip IC if no updates parameter to Yes.
   - To control the maximum number of days that can elapse before the Conditional Image Copy feature allows a scheduled batch image copy to be performed (regardless of whether database updates have occurred since the last image copy was performed), set the Perform IC if "nn" days since last IC parameter to the number of days.

5. Close the Maintain Parameters window, and click Yes to confirm that you want to save your changes.

To set the SMARTIC keyword

1. To enable the Conditional Image Copy feature, perform either of the following actions:
   - In a global options module that applies to the Image Copy utility job step, enter Y in the Conditional Image Copy field.
   - In the ICPSYSIN control statement data set, specify the SMARTIC(Y) keyword.
Using the Conditional Image Copy Simulation utility

The Conditional Image Copy Simulation utility (program ICPVSMIC) can help you estimate the extent to which the Conditional Image Copy feature can benefit your organization.

How the ICPVSMIC utility works

The ICPVSMIC utility scans an IMS RECON data set (or a VSAM copy of the RECON).

The utility identifies all data set groups (DSGs) and partitions for which both of the following conditions are true:

- The object has not been updated since the last image copy of the object.
- The latest image copy for the object is more recent than a defined number of days.

**Note**

When the ICPVSMIC utility is attempting to find possible candidates for the Conditional Image Copy feature, the utility rejects the following items:

- Items that are marked as IC-needed
- Items that are marked as nonrecoverable
- Items have been reorganized since the last image copy

The utility writes a report to identify these objects. If the Conditional Image Copy feature is implemented, the BMC Image Copy utility would bypass image copies for these objects. For more information, see “Using the Conditional Image Copy feature” on page 390.

Coding JCL to execute the ICPVSMIC utility

The following figure shows JCL for executing the ICPVSMIC utility.

**Figure 80: JCL to execute the Conditional Image Copy Simulation utility**

```plaintext
//ICPVSMIC JOB (ACCT), 'COND-IC SIM', MSGCLASS=X, CLASS=A
//ICPVSMIC EXEC PGM=ICPVSMIC,
//         PARM='nnn, your.recon.data.set.name'
//STEPLIB DD   DISP=SHR, DSN=product.load
//SYSOUT DD   SYSOUT=*  
```

For the PARM keyword on the EXEC statement, specify the following values:
Specify a numeric value (1 through 999) that represents a maximum number of days that should be allowed to elapse since the last image copy of a DSG or partition. The Conditional Image Copy Simulation utility uses this value to determine whether the image copy needs to be taken (regardless of updates) because the last image copy is as old as (or older than) the specified maximum.

Specify the data set name of an active IMS RECON data set, or specify the data set name of a VSAM copy of the RECON.

The ICPVSMIC utility requires APF authorization. You can ensure that all data sets in the STEPLIB are APF authorized, or you can ensure that a DBU Subsystem (DBUSS) or CPC address space is active on the z/OS system in which the ICPVSMIC utility is executing.

The recommended minimum region size is 6500 KB.

Example
You specify PARM='7,...' so that the utility considers seven days as the maximum number of days that is allowed to elapse since the last image copy.

- Database A has not been updated since the last image copy was taken two days ago. The utility indicates that the image copy for this database would be bypassed.

- Database B has not been updated since the last image copy was taken eight days ago. Because the last image copy is older than the specified value, the utility indicates that the image copy for this database would not be bypassed.

- Database C was updated two days ago, and the last image copy was taken five days ago. Because the database has been updated since the last image copy, the utility indicates that the image copy for this database would not be bypassed.

Using the ICPVSMIC report

The following figure shows an example of the report that the ICPVSMIC utility produces.

Figure 81: Conditional Image Copy Simulation utility report

<table>
<thead>
<tr>
<th>DSG/PART</th>
<th>DSG NAME</th>
<th>IMAGE COPY TIME</th>
<th># OF DAYS SINCE IC</th>
<th>CONDITIONAL IC</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP4P3</td>
<td>IP4P3N</td>
<td>2003.323 21:02:20.688480</td>
<td>2</td>
<td>YES</td>
</tr>
<tr>
<td>IP4X1</td>
<td>IP4X1A</td>
<td>2003.323 21:02:21.920950</td>
<td>2</td>
<td>YES</td>
</tr>
</tbody>
</table>
The report provides the following information:

- DSG or partition name, and DD name, of all DSGs and partitions that have not been updated since the last image copy

- Date and time of the last image copy for the DSG or partition

- Number of days that have elapsed since the last image copy of the DSG or partition

- Flag (YES) indicating that a requested image copy would be bypassed because the maximum number of days since the last image copy has not yet been reached

In the example report, EXEC PARM='8, your.recon.data.set.name' was specified; therefore, image copies would be bypassed for DSGs and partitions with image copies that are more recent than 8 days old.

- Number of DSGs and partitions for which image copies would be bypassed, and number of DSGs and partitions that have not been updated since the last image copy

Using the Conditional Reorg feature

The Conditional Reorg feature of Database Advisor saves time and resources by reorganizing only databases that are flagged as requiring reorganization.

Note

The Conditional Reorg feature applies to full-function databases and high availability large databases (HALDBs) only; it does not apply to Fast Path areas.
How the Conditional Reorg feature works

When a reorganization job is submitted, Database Advisor determines whether the
database is on the Exception List or whether it is forecast to be on the Exception List
before the next reorganization opportunity.

The next reorganization opportunity is the user-specified value in the Allow Reorg
field. The default value is 14 days.

If the database is on (or forecast to be on) the Exception List, reorganization proceeds
as scheduled.

If Database Advisor determines that a database does not require reorganization
(because an exception has not occurred and is not forecast), Database Advisor
terminates the job step with a user-specified completion code.

The Conditional Reorg feature can also run in simulate mode so that you can test the
results that would occur if the feature were activated. The Conditional Reorg History
report shows the results of active and simulated conditional reorganizations so that
you can view the resource savings that result (or would result) from the Conditional
Reorg feature.

MAXM/EP utilities versus Classic database reorganization
utilities

If you are running BMC database reorganization utility products that are based on
MAXM Extended Performance (MAXM/EP) architecture, the Conditional Reorg
feature runs automatically in Database Advisor (unless the feature is not activated).

The following BMC products and functions are based on MAXM/EP architecture:

- MAXM Reorg/Online for IMS
- MAXM Reorg/EP for IMS
- MAXM Reorg/EP for IMS with Online/Defrag Feature
- MAXM Reorg/EP Express for IMS
- UNLOAD PLUS/EP
- LOADPLUS/EP
- SECONDARY INDEX UTILITY/EP
- FAST REORG FACILITY/EP
- CHANGE RECORDING FACILITY
- PREFIX RESOLUTION PLUS
If you are running BMC database reorganization utility products that are based on "Classic" architecture, conditional reorganization is implemented as a stand-alone utility. The stand-alone utility performs the same Conditional Reorg function as is performed with Database Advisor, but the stand-alone utility runs as a separate job step.

The following BMC products and functions are based on Classic architecture:

- MAXM Reorg for IMS
- UNLOAD PLUS
- LOADPLUS
- SECONDARY INDEX UTILITY
- FAST REORG FACILITY

### Enabling the Conditional Reorg feature with the console

Enabling the Conditional Reorg feature involves activating the feature and specifying related parameters.

Database Advisor is distributed with the Conditional Reorg feature activated in simulate mode. In simulate mode, Database Advisor performs the same analysis and reporting functions as if the Conditional Reorg feature were activated but does not bypass any scheduled reorganizations.

### Activating the Conditional Reorg feature

You can activate the Conditional Reorg feature at the following levels:

- Host
- IMSPLEX
- Database group
- Database (full-function or DEDB)
- Partition (HALDB)

#### To activate the Conditional Reorg feature

1. In the Navigation window, right-click the object for which you want to activate the Conditional Reorg feature and choose **Configuration => Maintain Parameters**.
2. In the Maintain Parameters window, select Conditional Reorg in the parameter list.
Conditional Reorg parameters are displayed in the Maintain Parameters window (Figure 82 on page 400).

Figure 82: Maintain Parameters window

3 In the **Activate Conditional Reorg** feature, perform one of the following steps:
   - To activate the feature in normal mode, select **Yes**.
   - To activate the feature in simulate mode, select **Simulate**.

4 Click **Save**.

**Specifying Conditional Reorg parameters**

You can edit Conditional Reorg parameters in the Maintain Parameters window.

**To specify Conditional Reorg parameters**

1 In the Navigation window, right-click the object for which you want to specify Conditional Reorg parameters and choose **Configuration => Maintain Parameters**.

2 In the Maintain Parameters window, select **Conditional Reorg** in the parameter list.

3 Specify parameters.

   For more information about each parameter, including descriptions, valid values, and suggested defaults, see the online Help.
Using the stand-alone Conditional Reorg utility

The stand-alone Conditional Reorg utility performs the same function as the Conditional Reorg feature that runs automatically for MAXM/EP functions.

The stand-alone Conditional Reorg utility runs as a separate job step ahead of the database reorganization step that checks the condition code from the Conditional Reorg utility step.

Coding required JCL

To run the stand-alone Conditional Reorg utility, specify JCL as shown in the following figure.

**Figure 83: Stand-alone Conditional Reorg utility JCL**

```plaintext
/*
    /* CONDITIONAL REORG STEP
    /*
    //BMCCOND EXEC PGM=BMCCOND
    //STEPLIB DD DISP=SHR,DSN=YOUR.ADVISOR.LOADLIB
    //PLUSIN DD *
    ADVISE DBD(YOURDBD) RECON1(YOUR.RECON1.DSNAME)
    /*
    /* BYPASS THE REORG STEP IF NO REORG IS NEEDED FOR YOURDBD
    /*
    //REORG  EXEC BMCREORG,COND=(O,NE,BMCCOND)
*/
```

**Note**

If you are running classic database utilities that perform a complete reorganization, the JCL generator in the Console automatically includes Conditional Reorg JCL. The parameters can be specified on the EXEC statement or in the PLUSIN SYSIN data set.

Coding the ADVISE command and keywords

The ADVISE command invokes the Conditional Reorg utility.

Table 47 on page 402 lists keywords for the ADVISE command.
### Table 47: ADVISE command keywords for the stand-alone Conditional Reorg utility

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted value</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BYPASS</td>
<td>YES, Y, NO, N</td>
<td>NO</td>
<td>Specify whether to end the job with a completion code of zero without checking the status of the database.</td>
</tr>
<tr>
<td>CPCSSID</td>
<td>Subsystem name</td>
<td>CPCCOPT name that is specified in CPCCOPT member</td>
<td>Specify the name of the subsystem to use.</td>
</tr>
<tr>
<td>DBD</td>
<td>DBD name</td>
<td>None</td>
<td>Specify the DBD name of the database on which to run conditional reorganization.</td>
</tr>
<tr>
<td>PART</td>
<td>Single HALDB partition name</td>
<td>None</td>
<td>Specify the single HALDB partition name on which to run conditional reorganization (partitioned databases only).</td>
</tr>
<tr>
<td>RECON1</td>
<td>RECON1 data set name</td>
<td>RECON1 data set name that is specified in RECON1 DD statement or DFSMDA member</td>
<td>Specify the name of the RECON1 data set in which the database is registered with DBRC.</td>
</tr>
<tr>
<td>USERRC</td>
<td>Y, N</td>
<td>None</td>
<td>Specify where the utility obtains the return code for which it checks:</td>
</tr>
<tr>
<td></td>
<td>• Y—The keyword value uses the return code that is defined in the completion code for the DBD or the partition.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• N—The keyword value uses the condition code from the database reorganization step.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Handling error conditions

The following table lists error conditions for the Conditional Reorg utility.

You can use the JES viewer ("Viewing jobs in the console" on page 372) to view processing messages, including error conditions, in the job output.

Table 48: Conditional Reorg utility error conditions

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>Invalid JCL PARM string</td>
</tr>
<tr>
<td>002</td>
<td>Invalid ADVISE command</td>
</tr>
<tr>
<td>003</td>
<td>DBD name was not specified</td>
</tr>
<tr>
<td>004</td>
<td>RECON1 data set name was not specified</td>
</tr>
<tr>
<td>666</td>
<td>Program initialization error</td>
</tr>
</tbody>
</table>

If DBD and PART parameters are not specified in the PLUSIN statement, the utility generates a report listing the REORG status of all DBDs that are associated with the RECON.

Interpreting return codes

This table lists return codes that the stand-alone Conditional Reorg utility can issue when processing a single database.

Table 49: Return codes for the Conditional Reorganization utility

<table>
<thead>
<tr>
<th>Return code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>The database is registered and needs reorganization.</td>
</tr>
<tr>
<td>02</td>
<td>The database is registered, but no data is available for a Conditional Reorg check.</td>
</tr>
<tr>
<td>04</td>
<td>The database is not registered.</td>
</tr>
<tr>
<td>08</td>
<td>The database is registered but does not need reorganization.</td>
</tr>
<tr>
<td>10</td>
<td>The database is registered, the database does not need reorganization, and the utility is running in simulate mode.</td>
</tr>
</tbody>
</table>
Using the Conditional Reorg History report

The Conditional Reorg History report shows information about Conditional Reorg execution. From the report you can learn the resource savings that you have gained by having the Conditional Reorg feature bypass unnecessary reorganization. The report is populated when the Conditional Reorg feature is activated or is in simulate mode.

You can view the Conditional Reorg History report at the following levels:

- Host
- IMSPLEX
- Database (full-function or DEDB)
- Partition (HALDB)

To view the Conditional Reorg History report

1. In the Navigation window, right-click the object for which you want to open the Conditional Reorg History report and choose Reports => Conditional Reorg History.
The report is displayed in the Conditional Reorg History window (Figure 84 on page 405).

Figure 84: Conditional Reorg History window

For more information about the Conditional Reorg History report, see the online Help.

Using the Conditional Reorg batch report

The Conditional Reorg batch report is generated by evaluating all databases that are associated with a RECON and by calculating whether a database needs to be reorganized.

The Conditional Reorg batch report runs by using the settings defined in Conditional Reorg parameters. However, the Conditional Reorg batch report does not take any action. For more information about the Conditional Reorg feature, see “Enabling the Conditional Reorg feature with the console” on page 399.
Coding required JCL

To run the Conditional Reorg batch report, specify JCL as shown in the following example.

**Figure 85: Conditional Reorg batch report JCL**

```
//STEP1 EXEC PGM=BMCCOND
//STEPLIB DD DISP=SHR,DSN=BMC.LOAD
//*
//PLUSIN DD *
ADVISE RECON1(YOUR.RECON1) -
BYPASS(N) CPCSSID(CPCZ)
/*
```

The ADVISE command invokes the Conditional Reorg batch report. For a description of the keywords for the ADVISE command, see “Coding the ADVISE command and keywords” on page 401.

If you do not specify the DBD and PART parameters in the PLUSIN statement, the report includes all DBDs that are associated with the RECON.

Conditional Reorg batch report

The following figure shows the Conditional Reorg batch report.

**Figure 86: Conditional Reorg batch report**

```
10:30:46.26 DBDNAME:DTDAA10P - DATABASE DOES NOT NEED A REORG AT THIS TIME /
CONDITIONAL REORG BYPASSED, SIMULATE MODE
10:30:46.35 DBDNAME:DTDAA12P - DATA BASE REORG STATUS IS NOT AVAILABLE
10:30:46.49 DBDNAME:DTDAA13P - DATA BASE REORG STATUS IS NOT AVAILABLE
10:30:46.60 DBDNAME:DUMMYDB - DATABASE DOES NOT NEED A REORG AT THIS TIME /
CONDITIONAL REORG BYPASSED, SIMULATE MODE
10:30:46.71 DBDNAME:YBX0DA PARTITION NAME:YBX0P1 - DATA BASE NEEDS A REORG
10:30:46.94 DBDNAME:HD0FMO3 - DATA BASE NEEDS A REORG
10:30:47.05 DBDNAME:HD0FMO5 - DATA BASE NEEDS A REORG
```

Table 50 on page 406 describes status values that are returned by the Conditional Reorg batch report.

**Table 50: Conditional Reorg batch report status values**

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA BASE DOES NOT NEED A REORG AT THIS TIME / CONDITIONAL REORG BYPASSED, SIMULATE MODE</td>
<td>Based on your parameters, the database does not require (or will not require) a reorganization before the next scheduled reorganization opportunity.</td>
</tr>
<tr>
<td>DATA BASE REORG STATUS IS NOT AVAILABLE</td>
<td>The database is defined to Database Advisor, but no data has been captured for the database.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DATA BASE NEEDS A REORG</td>
<td>Based on your parameters, the database requires (or will require) a reorganization before the next scheduled reorganization opportunity.</td>
</tr>
</tbody>
</table>

### Error conditions

The Conditional Reorg batch report shows error conditions for the Conditional Reorg utility.

For a list of error conditions for the Conditional Reorg batch report, see “Handling error conditions” on page 403. You can use the JES viewer (see “Viewing jobs in the console” on page 372) to view processing messages, including error conditions, in the job output.
Monitoring recovery job steps

This section provides information about the Monitor Recovery function.

Overview of the Monitor Recovery function

This section provides general information about the Monitor Recovery function.

Purpose of the Monitor Recovery function

The Monitor Recovery function receives information about recovery processing from the BMC Recovery utility and displays this information in the Monitor Recovery window and dependent windows in the console.

You can use the Monitor Recovery function to watch the progress of recovery job steps, to determine whether a problem has occurred, and to check on the recovery of objects. The function helps you answer the following types of questions:

- What is happening with my recovery job steps? Are they executing as expected? What phase (such as reading logs, merging records, or rebuilding indexes) are they in?
- Did a warning or failure occur during a recovery job step? If so, which objects had failures? What were the return codes?
- Is a particular object (data set or area) included in a recovery job step? What is the status of an object?

Graphical progress bars and charts provide an intuitive, overall view of the current recovery environment. Distinctive icons that indicate problems help you recognize those problems quickly so that you can get right to work on solving them. You can select items and drill down for more information.

The Monitor Recovery function is helpful any time that you want to track a recovery. The function is especially suited to disaster recovery scenarios when a large number
of recoveries are executing simultaneously. With icons to highlight problems, problems are less likely to be overlooked than they are in a traditional interface.

The graphical view of the recovery environment can help disaster recovery team members communicate with each other, with managers, and with users faster and more effectively. Moving progress bars and green status icons can reassure you that the disaster recovery process is working as it should.

**How the Monitor Recovery function works**

The Monitor Recovery function works with the BMC Recovery utility to collect and report recovery information.

During initialization of a recovery job step, the Recovery utility tries to connect to the Cross Product Connectivity (CPC) server. If the server is not available or the Monitor Recovery function is not licensed for use, the recovery job step proceeds as normal and does not try to provide information to the Monitor Recovery function. The function cannot report information about this job step.

If the server is available and the Monitor Recovery function is licensed for use, the Recovery utility notifies the server that the recovery job step is in progress. When you select the Monitor Recovery function in the console, the console sends a request to the server to obtain recovery information for the selected RECON. The server returns a list of all recovery job steps that are executing or that have been updated within the time period that you specify.

The Recovery utility posts detailed information about the recovery job step in a location that the Monitor Recovery function can access independently on demand. Your console session automatically requests a periodic refresh of job step and activity information. You can request a manual refresh of failure and object information. You can adjust the automatic refresh rate for your session without affecting the refresh rate for other user sessions.

If you set the Recovery Failures Email Notification parameter to **Yes**, the Monitor Recovery function generates a notification message if a problem occurs during the recovery job.

---

**Note**

Performance of the recovery job step is not degraded by using the Monitor Recovery function. The rate at which the Recovery utility posts information about the job step depends on performance of the job step (which is affected by factors such as the dispatching priority). The number of users who are using the Monitor Recovery function and the rates at which they request a console refresh have no effect on recovery job step performance.
Considerations for the Monitor Recovery function

The following considerations apply to the Monitor Recovery function:

- To access the Monitor Recovery function in the console, you must have a license for the Backup and Recovery Solution for IMS product. If you do not have a license, the function is dimmed in console menus.

- For a recovery job step to be detected by the Monitor Recovery function and to be displayed in the Monitor Recovery window, the recovery job step must be running with a license for Backup and Recovery Solution for IMS.

- The UIM server, the BCSS/CPC server, and the ADV server must be installed and active, as described in “Installing, configuring, and starting mainframe components” on page 72.

- You must install the console on your Windows system, as described in “Installing the console” on page 76.

- Your IMS environment must be identified to Advisor by configuring the IMS system in the console, as described in “Defining your IMS database administration environment” on page 115.

Working with the Monitor Recovery function

This section explains how to access, customize, and use the Monitor Recovery function.

Monitor Recovery function windows

You work with the Monitor Recovery function through a set of windows in the console.
The Monitor Recovery window (Figure 87 on page 412) is the primary window for the function. Dependent secondary windows (the View Activity window, the View Failures window, and the View Objects window) provide more information.

Figure 87: Monitor Recovery window

You can customize the behavior of the Monitor Recovery window and dependent windows as explained in “Setting options for Monitor Recovery function windows” on page 414.

The Monitor Recovery window provides the following major features:

- The Jobs List table is a central, concise view of the recovery job steps that are executing now or that have executed within the specified time period. For more information, see “Working with the list of recovery job steps” on page 416.

- The Job Details area provides in-depth information about job steps that you select in the Jobs List. For more information, see “Interpreting information in the Job Details area” on page 416.
The Object Phase Summary area provides a graphical view of the phases of processing that are in progress for a selected job step. For more information, see “Interpreting information in the Object Phase Summary area” on page 417.

From the Monitor Recovery window, you can access the following dependent windows:

- The View Activity window displays more information about log input processing and active objects. For more information, see “Viewing job step activity information” on page 418.

- The View Failures window displays more information about the failures that have been detected during execution of a selected job step. For more information, see “Viewing job step failure information” on page 422.

- The View Objects window displays more information about the objects that are being processed or that were processed during execution of a selected job step. For more information, see “Viewing job step object information” on page 424.

Accessing the Monitor Recovery function

You can access the Monitor Recovery function in the Navigation window.

1. Perform one of the following steps:

   - Right-click an IMS RECON object in the Navigation window, and choose Monitor Recovery from the pop-up menu.

   - Left-click an IMS RECON object in the Navigation window, and choose Action => Monitor Recovery from the menu bar.

The Monitor Recovery window (Figure 87 on page 412) is displayed

Note

If the Monitor Recovery window is already displayed in the work area and you right-click a different IMSPLEX in the Navigation window and choose Monitor Recovery, another instance of the Monitor Recovery window is displayed for the second object. When you select a different object, the information in the original Monitor Recovery window is not affected.

When you close the Monitor Recovery window, all dependent windows are also closed.
Setting options for Monitor Recovery function windows

You can set options to customize the behavior of the Monitor Recovery function windows.

These settings apply to your own console session and have no effect on other user sessions.

Pausing and resuming automatic refreshes

When the Monitor Recovery window or the View Activity window is active, the information is automatically refreshed. When the window is inactive, no automatic refresh occurs.

To pause and resume automatic refresh

1 To pause (freeze) the information that is displayed in the window, click Pause.

   The pause applies to the active window only; it does not apply to other Monitor Recovery function windows.

2 When ready to resume automatically refreshing this window, click Resume.

Controlling the automatic refresh rate

You can change the rate at which the Monitor Recovery function performs an automatic refresh of the recovery information on the Monitor Recovery window and in the View Activity window. The same refresh rate applies to both windows.

The fields at the top right side of the Monitor Recovery window and the View Activity window display the current refresh rate and the date and time when the last refresh occurred.

To control the automatic refresh rate

1 Click Set Refresh Rate.

   The Set Refresh Rate dialog box is displayed.

2 In the Refresh rate field, type a value (6 through 100) to indicate the number of seconds between requests for refreshed information. The default value is 6 seconds.

3 Click OK.
**Manually refreshing information**

Information in the View Failures window and the View Objects window is not refreshed automatically; in these windows, you can click **Refresh** to refresh the information manually.

In the View Failures window and the View Objects window, the top right side displays the date and time that the last refresh occurred.

**Controlling the time period for listing job steps**

You can change the time period for listing recovery job steps on the Monitor Recovery window.

**To control the time period for listing job steps**

1. Enter a value (1 through 96) in the **Show jobs within the last n hour(s)** field.

   The default value is 8 hours.

   The Monitor Recovery function returns information about all recovery job steps that are currently executing or that have been updated within the time period that you specify.

   For example, to list all recovery job steps that are active or have been active within the last day, select 24. To list all recovery job steps that are active or have been active within the last hour, select 1.

**Accessing more information through the JES viewer**

If you require more details for problem analysis, you can use the JES viewer through the console.

For more information about the JES viewer, see “Viewing jobs in the console” on page 372.

The Monitor Recovery function provides shortcuts to the JES viewer.

**To access more information through the JES viewer**

1. On the View Failures window or the View Objects window, click **JESViewer**.

   The JES Viewer window is displayed.
Working with the list of recovery job steps

The Jobs List table of the Monitor Recovery window provides a central, concise view of the recovery job steps that are executing or that have been updated within the specified time period.

Each row in the table represents a job step.

For each job step, a progress bar graphically represents progress of the job step toward completion and states the percentage of objects for which recovery is complete. If logs tasks are present in the recovery job step, 20 percent of the displayed progress is attributed to log tasks. The remaining 80 percent of the displayed progress is attributed to recovery tasks. If no logs tasks are present, 100 percent of the activity is attributed to recovery tasks.

For each job step, a status icon indicates the most severe status that the job step has experienced. Table 51 on page 416 lists the possible statuses in order from least severe to most severe.

Table 51: Monitor Recovery status icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔</td>
<td>A green square containing a check mark indicates that all tasks are executing normally or have completed with return code 0.</td>
</tr>
<tr>
<td>🟢</td>
<td>A yellow triangle containing an exclamation point (!) indicates that one or more tasks have encountered warning conditions (return code 4).</td>
</tr>
<tr>
<td>✗</td>
<td>A red circle containing an X indicates that one or more tasks have encountered error conditions (return code 8 and higher), but the job step has not yet ended. When this job step ends, the red circle changes to a black circle.</td>
</tr>
<tr>
<td>❌</td>
<td>A black circle containing an X indicates that one or more tasks have encountered error conditions (return code 8 and higher) and the job step has ended.</td>
</tr>
</tbody>
</table>

You can sort the columns in the Jobs List table by clicking the column heading. The columns are displayed initially in ascending order by job step name.

You can highlight (by clicking or right-clicking) a row to obtain more information about the job step that is listed in that row.

Interpreting information in the Job Details area

When you highlight a row in the Jobs List table, the Monitor Recovery window displays summary information about the selected job step in the Job Details area. (At initial display, the Monitor Recovery window contains details about the first job step in the Jobs List).
The Job Details area includes the following types of information:

- Job name, JES job number, and job step name
- Status of the job step (as explained in Table 52 on page 417)
- Current overall processing phase of the recovery job step
  
  As soon as the job step starts the processing of a phase for any object, the overall phase is switched to that phase. Therefore, the current phase field reports the latest phase that applies to any object. The phases are explained in Table 52 on page 417.

- Start time and elapsed time of the job step
- Amount of CPU time that the job step has consumed

For more information about the information in the Job Details area, see the online Help.

<table>
<thead>
<tr>
<th>Job phase</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduled</td>
<td>At least one object is included in the list of objects to be recovered or rebuilt, but processing has not yet started for any object.</td>
</tr>
<tr>
<td>Log Input</td>
<td>The Recovery utility is reading and sorting log records for at least one object.</td>
</tr>
<tr>
<td>Merge</td>
<td>The Recovery utility has identified the necessary input (image copy, change accumulation, and/or log) data sets to recover at least one object and is merging the records from those data sets to recover the object.</td>
</tr>
<tr>
<td>Completed</td>
<td>Recovery or rebuild processing for all objects has completed.</td>
</tr>
</tbody>
</table>

Interpreting information in the Object Phase Summary area

The Object Phase Summary area of the Monitor Recovery window provides a graphical view of the phases of object processing that are in progress for the job step that is selected in the Jobs List table.

The pie chart represents the total number of objects that the Recovery utility is recovering or rebuilding in this job step. (Object size is not considered.) Each slice of the "pie" represents the number of objects that are in the indicated phase (or stage) of processing. Table 53 on page 418 lists the object phases in order from earliest to latest.
Table 53: Monitor Recovery information in the Object Phase Summary area

<table>
<thead>
<tr>
<th>Object Phase</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduled</td>
<td>Red</td>
<td>The indicated number of objects are included in the list of objects to be recovered or rebuilt, but the recovery or rebuild task has not yet started for the objects.</td>
</tr>
<tr>
<td>Log Input</td>
<td>Blue</td>
<td>The Recovery utility is reading and sorting log records for the indicated number of objects.</td>
</tr>
<tr>
<td>Merge</td>
<td>Green</td>
<td>The Recovery utility has identified the necessary input (image copy, change accumulation, and/or log) data sets and is merging the records from those data sets to recover the indicated number of objects.</td>
</tr>
<tr>
<td>Build Index</td>
<td>Yellow</td>
<td>The Index Rebuild function is rebuilding the indexes for the indicated number of objects.</td>
</tr>
<tr>
<td>Ended</td>
<td>Gold</td>
<td>The recovery or rebuild task for the indicated number of objects is no longer executing. The task has completed normally or has terminated with errors.</td>
</tr>
</tbody>
</table>

Viewing job step activity information

You can use the View Activity window to view information about the activities that are occurring or that have occurred during execution of a selected job step.
In addition to customization options (which are described in “Setting options for Monitor Recovery function windows” on page 414), the window contains the following tabs:

- The Log Input tab contains information about the log data sets that the Recovery utility is processing or has processed. For more information, see “Interpreting information about input logs” on page 420.

- The Active Objects tab contains information about the objects that the Recovery utility is processing. For more information, see “Interpreting information about active objects” on page 420.

**Accessing job step activity information**

You can view the job step activity information in the View Activity window.

**To access the View Activity window**

1. Perform one of the following steps:
Right-click the row for a job step in the Jobs List on the Monitor Recovery window, and choose **View Activity** from the pop-up menu.

Highlight the row for a job step in the Jobs List on the Monitor Recovery window, and click **View Activity** at the top of the Jobs List.

The view activity window (Figure 88 on page 419) is displayed.

### Interpreting information about input logs

The Log Input tab is displayed when you first access the View Activity window if the job step is still processing logs.

On this tab, the log file table provides information about each log data set that the Recovery utility is processing or has processed during this job step. The following information is included:

- Data set name of the log data set
- The time when utility started reading the log data set
- The time when utility finished reading the log data set
- Number of bytes and records that the Recovery utility has read from the log
- Number of bytes and records that the Recovery utility has passed to the sort facility
- Amount of time that has elapsed during processing of the log

Below the log file table, the **Log files read** field contains a progress bar that indicates the percentage of the total number of log files that the Recovery utility has processed during this job step. The **Passed to sort** field contains the total number of bytes and records that the Recovery utility has passed to the sort facility during this job step.

You can sort the table on a field by clicking the column heading.

For more information about the information in this tab, see the online Help.

### Interpreting information about active objects

The Active Objects tab is displayed when you first access the View Activity window if the job step has completed all log processing.
On this tab, a table provides information about each object that the Recovery utility is processing.

Figure 89: View Activity window (Active Objects tab)

When processing is complete for an object, the object is removed from the table. The following fields are included:

- DBD name, partition or area name, and data set group name of the object
- Object type (such as HIDAM, secondary index, or partitioned HDAM)
- Percentage of units (blocks for OSAM objects, CIs for VSAM ESDS objects, or records for VSAM KSDS objects) for which processing is complete
- Total number of units (blocks for OSAM objects, CIs for VSAM ESDS objects, or records for VSAM KSDS objects) to be processed for the object
- Start time of the task that is processing the object
- Processing phase of the active object, as described in Table 54 on page 422
- Amount of time that has elapsed time during processing of the object
- Amount of CPU time that has been consumed to process the object
You can sort the table on a field by clicking the column heading.

For more information about the information in this tab, see the online Help.

**Table 54: Monitor Recovery object phases**

<table>
<thead>
<tr>
<th>Object phase</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduled</td>
<td>The object is included in the list of objects to be recovered or rebuilt, but processing has not yet started for the object.</td>
</tr>
<tr>
<td>Log Input</td>
<td>The Recovery utility is reading and sorting log records for the object.</td>
</tr>
<tr>
<td>Restore</td>
<td>The Recovery utility is restoring the object from a copy.</td>
</tr>
<tr>
<td>Merge</td>
<td>The Recovery utility has identified the necessary input (image copy, change accumulation, and/or log) data sets and is merging the records from those data sets to recover the object.</td>
</tr>
<tr>
<td>Build Index</td>
<td>The Index Rebuild function is rebuilding the index object.</td>
</tr>
<tr>
<td>Completed</td>
<td>Recovery or rebuild of the object was successful.</td>
</tr>
<tr>
<td>Not Scheduled</td>
<td>The Recovery utility abandoned processing of an object because a failure occurred before the recovery task started. For example, the utility can abandon processing of an index because a failure occurred during recovery of the primary database.</td>
</tr>
</tbody>
</table>

**Viewing job step failure information**

You can use the View Failure window to view information about the failures (tasks that have completed with a return code greater than 0) that have occurred during execution of a selected job step.
In addition to customization options (see “Setting options for Monitor Recovery function windows” on page 414), the window contains a table that provides detailed information about the objects for which failures have occurred. For more information, see “Interpreting information about failures” on page 423.

**Accessing failure information**

You can access the View Failures window using the Monitor Recovery window.

**To access failure information**

1. Perform one of the following steps:

   - Right-click the row for a job step in the Jobs List on the Monitor Recovery window, and choose View Failures from the pop-up menu.

   - Highlight the row for a job step in the Jobs List on the Monitor Recovery window, and click View Failures at the top of the Jobs List.

The View Failures window (Figure 87 on page 412) is displayed

**Interpreting information about failures**

The table in the View Failures window provides information about the objects for which failures have occurred during this job step.

The following information is included:
- DBD name, partition or area name, and data set group name of the object
- Object type (such as HIDAM, secondary index, or partitioned HDAM)
- Highest return code of the task that was processing the object
- Start time and end time of the task that was processing the object

You can sort the table on a field by clicking the column heading.

For more information about the information in this window, see the online Help.

For more information about failures, you can click JESViewer to display the JES viewer. For more information about the JES viewer, see “Viewing jobs in the console” on page 372.

**Viewing job step object information**

You can use the View Objects window to view information about all (or selected) objects that have been (or are being) processed during execution of a selected job step.

*Figure 91: View Objects window*

In addition to customization options (see “Setting options for Monitor Recovery function windows” on page 414), the window contains a table that provides detailed information about the objects that have been (or are being) processed by this job step. For more information, see “Interpreting information about objects” on page 425.
**Accessing object information**

You can view objects on the Monitor Recovery window.

1. Perform one of the following steps:
   - Right-click the row for a job step in the Jobs List on the Monitor Recovery window, and choose **View Objects** from the pop-up menu.
   - Highlight the row for a job step in the Jobs List on the Monitor Recovery window, and click **View Objects** at the top of the Jobs List.

   The View Objects window ([Figure 91 on page 424](#)) is displayed.

**Filtering the list of objects**

When you first access the View Objects window, the Filter Objects dialog box is displayed automatically so that you can limit the list to objects that interest you.

You can change the filter at any time after the View Objects window is displayed by clicking **Filter**.

To filter the list of objects, enter the full or partial name of the DBD, the full or partial name of the partition or area, or both. If you enter both, the object must match both fields to be displayed in the list.

To list all objects, you can leave these fields blank.

**Figure 92: Filter Objects dialog box**

![Filter Objects dialog box](#)

**Interpreting information about objects**

The table in the View Objects window provides detailed information about the objects that have been (or are being) processed during this job step.

The following information is included:
Notifying users about recovery problems

To report problems that occurred during a BMC Recovery utility job step, the Monitor Recovery function can send email messages to specified addresses.

Enabling and controlling notification messages

To notify users about recovery problems, the Monitor Recovery function uses some of the Advisor e-mail notification infrastructure and parameter values that also support automated notification of exceptions.

1. From the Database Management console, right-click a host or RECON object in the Navigation window, and choose Configuration => Maintain Parameters.

2. On the left side of the Maintain Parameters window, select E-mail / WTO.

3. On the main E-mail / WTO Parameters branch, set the Recovery Failures Email Notification parameter to Yes.

4. On the Warning, Critical, and Dead branches, set the Warning/Critical/Dead e-mail alert option that you want to use to Yes, and specify e-mail addresses to receive messages:

- To Warning addresses, the Monitor Recovery function sends information about any task that ended with a return code greater than 0. These addresses potentially receive the largest amount of information.
To Critical addresses, the Monitor Recovery function sends information about any task that ended with a return code greater than or equal to 8.

To Dead addresses, the Monitor Recovery function sends information about any task that abnormally terminated with a system or user abend. These addresses receive information about the most severe problems.

Note

The Monitor Recovery function ignores the Issue WTO’s parameter and the Consolidate E-mail Messages parameter. The function always sends the complete report at the end of the Recovery utility job step. You can obtain WTO messages by using features and options of the Recovery utility.

For more information, see “E-mail / WTO / REXX EXEC parameters” on page 253.

Content of notification messages

The notification messages from the Monitor Recovery function report tasks that encountered problems during the Recovery utility job step.

The message lists a task if the return code for that task has the minimum severity that corresponds with the Warning, Critical, or Dead parameter. For an example message, see Figure 93 on page 427.

Figure 93: E-mail message from Monitor Recovery function

The message contains the following fields:

- The heading identifies the job step name, and the date and time when the job step started. (The subject line of the e-mail message also provides this information.)

- For each database data set that the Recovery utility was processing when a problem occurred, the left side of the report lists the DBD name, area or partition name, ddname, and type of database organization.

- The right side of the report lists the task phase that was executing when the problem occurred. This side also reports the highest return or abend code that was set during the task, and when the task started and ended.
Using reports

This section contains information about the reports that are available in the console for the BMC database administration products for IMS. You can use reports to evaluate data states, research exceptions, and track data trends and utility executions.

Table 55 on page 429 lists reports that are displayed in the console when you choose Reports from the menu for an object. (BMC database administration products for IMS produce other statistical reports.) If you have Database Advisor, you can also request a batch version of some reports for an IMSPLEX; for more information, see “Requesting batch reports for an environment” on page 444.

Table 55: Reports

<table>
<thead>
<tr>
<th>Report</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area State</td>
<td>Provides data about the current state of a selected DEDB area</td>
<td>“Evaluating the current state of your IMS data” on page 430</td>
</tr>
<tr>
<td>Area History</td>
<td>Provides historical data about a selected DEDB area</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Assigned Databases</td>
<td>Lists databases that are assigned to a selected database group</td>
<td>Managing groups by using the Assigned Databases report on page 129</td>
</tr>
<tr>
<td>Compare History</td>
<td>Compares historical data for two or more selected objects</td>
<td>“Comparing historical statistics for multiple objects” on page 441</td>
</tr>
<tr>
<td>Conditional Reorg History</td>
<td>Provides historical data about Conditional Reorg utility executions</td>
<td>“Using the Conditional Reorg History report” on page 404</td>
</tr>
<tr>
<td>Data Set History</td>
<td>Provides historical data about a selected data set</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Data Set State</td>
<td>Provides data about the current state of a selected data set</td>
<td>“Evaluating the current state of your IMS data” on page 430</td>
</tr>
<tr>
<td>Database History</td>
<td>Provides historical data about a selected database</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Database State</td>
<td>Provides data about the current state of a selected database</td>
<td>“Evaluating the current state of your IMS data” on page 430</td>
</tr>
<tr>
<td>Report</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Exception Forecast</td>
<td>Lists exceptions that are forecast for a user-defined range of dates</td>
<td>“Using the Exception Forecast report” on page 168</td>
</tr>
<tr>
<td>HALDB History</td>
<td>Provides historical data about a selected HALDB</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>HALDB State</td>
<td>Provides data about the current state of a selected HALDB</td>
<td>“Evaluating the current state of your IMS data” on page 430</td>
</tr>
<tr>
<td>Job History</td>
<td>Provides historical data about utility executions for a host or IMSPLEX object</td>
<td>“Tracking utility executions” on page 442</td>
</tr>
<tr>
<td>Last Data Collection</td>
<td>Lists objects for which data has <em>not</em> been collected within a specified time frame</td>
<td>“Listing last data collection data for objects” on page 443</td>
</tr>
<tr>
<td>Partition History</td>
<td>Provides historical data about a selected partition</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Partition State</td>
<td>Provides data about the current state of a selected partition</td>
<td>“Evaluating the current state of your IMS data” on page 430</td>
</tr>
<tr>
<td>RECON State</td>
<td>Provides data about the current state of the active RECON data set</td>
<td>“Evaluating the state of RECONs” on page 434</td>
</tr>
<tr>
<td>Segment History</td>
<td>Provides historical data about a selected segment</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Space Usage History</td>
<td>Provides historical data about space usage for a selected data set</td>
<td>“Researching exceptions, and tracking trends” on page 437</td>
</tr>
<tr>
<td>Unassigned Databases</td>
<td>Lists databases in the IMSPLEX that are not assigned to a database group</td>
<td>“Managing groups by using the Unassigned Databases report” on page 131</td>
</tr>
</tbody>
</table>

**Evaluating the current state of your IMS data**

You can use state reports to evaluate the current condition of your IMS data.

*Note*

You can also use state reports to evaluate the condition of your RECONs. For more information, see “Evaluating the state of RECONs” on page 434.

**Overview of state reports for database-type objects**

State reports for database-type objects offer a detailed, comprehensive view of the current state of your IMS data.
The data in the state reports reflects the state of a database-type object when that data was most recently collected.

State reports include data forecasts. A data forecast is a graph that shows how close a current state value is to a threshold value and the rate at which the current state value is approaching the threshold.

State reports display data in graphs and tables, and you can export tabular data for use in third-party spreadsheets.

**Viewing state reports for database-type objects**

You can view state reports for database, HALDB, partition, area, and data set objects.

Steps that you perform to view any state reports are virtually the same. The only difference is the name of the report that you select from the Reports menu.

**To view a state report**

1. In the Navigation window, right-click the object for which you want to view a state report and choose **Reports => reportname**.
The state report for the selected object is displayed. Figure 94 on page 432 is an example of a Data Set State report.

**Figure 94: Data Set State report**

All state reports for database-type objects include a Properties tab and a Forecast tab. Other tabs are specific to the report that you are viewing. Some tabs have subtabs that you can click. To navigate among tabs, click the tab corresponding to the type of information that you want to view.

**Displaying forecasts for database-type objects**

Database Advisor charts a forecast for the key elements that are collected and used during the Database Advisor Analyze process.

The charts show a graphical representation of the comparison of the current state of a key element to the threshold value that is defined for the key element.
The data forecast is available only if you have a license for Database Advisor.

The data forecast is available through the Forecast tab in all state reports. Figure 95 on page 433 is an example of the forecast for a data set object. You can click each chart in the forecast to enlarge the chart in a separate dialog box.

Figure 95: Forecast tab of the State report

Figure 96 on page 434 is an example of the dialog box that is displayed when you click a chart in a forecast chart. Information in the dialog box includes the exception
status, the base value, the current state value, the threshold value, and the dates on which the information was collected.

**Figure 96: Forecast detail dialog box**

---

**Evaluating the state of RECONs**

You can use RECON State reports to evaluate the current condition of your RECONs.
Overview of RECON State reports

The RECON State report identifies the RECON data sets that are designated as the COPY1 data set, the COPY2 data set, and the spare. The report provides comprehensive information about the current state of the RECON data sets.

The data in this report reflects the state of the RECON data sets when Recovery Advisor was last executed. If Recovery Advisor has not been executed for this RECON, the RECON State report is not available.

Viewing RECON State reports

You can view a RECON State report in the Navigation window.

1. Right-click an IMSPLEX object in the Navigation window and choose Reports => Recon State.

The RECON State report is displayed. For the active RECONs, the RECON State report offers data in chart form (Figure 97 on page 436). For all RECONs, the report offers data in tabular form (Figure 98 on page 437). You can export the data for analysis by a third-party spreadsheet program.
For more information about the RECON State report, see the online Help.

**Figure 97: RECON State report (RECONn Charts tab)**
Researching exceptions, and tracking trends

Database Advisor provides history reports that you can use to research exceptions and to track trends.

Overview of history reports

Examining past behavior of your IMS data can help you determine why a problem occurs.

You can use the historical data to understand why an exception was reported to the Exception List or to determine when and where an error occurred.

You can also use historical data to track data trends. Identifying patterns of behavior can help you plan for data management requirements. For example, you may notice
that at the end of every fiscal quarter a particular database runs short on space. By identifying this trend, you can take actions to avoid problems.

History reports provide much information about your IMS data. A set of statistics is collected and is organized into manageable reports.

History reports display data in graphs and tables that you can customize to control the type of data in the report. You can export tabular data for use in third-party spreadsheets.

**Viewing history reports**

You can view historical data in history reports for host, IMSplex, database, HALDB, partition, area, data set, and segment objects. The steps you perform to view any of the history reports are basically the same. The only difference is the name of the report that you select in the Reports menu.

1. In the Navigation window, right-click the object for which you want to view a history report and choose **Reports => reportname**.

The only history report that is not in the Reports menu is the Segment History report. Segment History (full function databases only) is the only menu command that is available at the segment level.
Most history reports include a Chart tab, a Table tab, a Last Execution tab, and an Options tab. Some tabs have subtabs that you can click. To navigate among the tabs, click the tab corresponding to the type of information that you want to view.

**Chart tab**

Graphical representations of your data are displayed on the Chart tab.

You can select the fields to display, the type of chart to use, and more chart options on the Options tab. The Chart tab offers a pop-up menu with menu commands that let you zoom, print, and save the graphic. To access the pop-up menu, right-click the chart.

**Table tab**

Tabular data is displayed on the Table tab.
You can select the type of information that you want to be displayed on the Options tab. For example, you may want to populate the table with monthly averages. You can also export the data for use in third-party spreadsheets.

**Last Execution tab**

The Last Execution tab displays data collected from the last utility execution that was run on the database (or selected object).

**Options tab**

You can use the Options tab to set view options for data displayed on Chart and Table tabs.

For more information about these options, see “Customizing history reports” on page 440.

**Customizing history reports**

For all history reports, you can set options to customize data that is displayed in the report and to determine how the data is displayed.

You can control the type of data that is used to populate reports. You can also select an option to display data for a specific data range.

You can use the charting options to control the parameters that are charted, the type of chart to be displayed (pie chart, bar chart, line graph), and (for bar charts) whether to chart values by parameter or by date.
Figure 100 on page 441 is an example of the Options tab that you can use to set options for viewing historical data.

**Figure 100: Database History report Options tab**

![Database History report Options tab](image)

For more information about setting options for history reports, see the online Help.

**Comparing historical statistics for multiple objects**

Compare History reports are similar to other history reports but show data for multiple databases, partitions, or areas. You can use these reports to visually compare statistics for two or more (up to five) objects in a chart or table format.

1. To access Compare History reports in the Navigation window, right-click the IMS Databases object, and choose **Reports => Compare History => Compare type History**.

   For more information, see the online Help.
Tracking utility executions

You can use the Job History report to display information about executions of utilities against objects in an IMSPLEX or a sysplex (at the host-connection level).

Overview of the Job History report

The Job History report provides historic information about utility jobs that have executed against IMS databases in the selected system.

Tabs in the Job History window provide:

- Graphical information about the jobs in chart form
- Detailed information about the jobs in tabular form
- Specific information about the last execution of a utility job in list form
- Options for changing the displayed information, including whether and how to display detailed or summary information, which parameter to display on charts, and whether and how to filter the information by a range of dates

You can export displayed tabular information to an external file.

From the Job History report you can determine the duration of the job, the CPU time, and the physical input/output (I/O) count.

---

**Note**

Job History information is not available for Fast Path DEDBs or for stand-alone Image Copy, Recovery, or Change Accumulation jobs.

Job History information is not available for Fast Path DEDBs. Job History information is not available for stand-alone BMC Image Copy, Recovery, or Change Accumulation utility jobs unless the JOBHIST(Y) option was specified in the global options or in the control statement data set for the job.

---

Viewing the Job History report

The Job History report is opened the same way as all other history reports.
For more information about viewing a Job History report (Figure 101 on page 443), see “Viewing history reports” on page 438.

Figure 101: Job History report

The Job History report provides the same customization options on the Options tab that are available for all other history reports. The Job History report also provides an option to filter the data by utility. For more information about customizing a history report, see “Customizing history reports” on page 440.

Listing last data collection data for objects

The Last Data Collection report shows information on when and how statistical data has been collected for the databases that are registered in the selected IMSPLEX (RECON). You can view the Last Data Collection report for IMSPLEX objects.

1. In the Navigation window, right-click the IMSPLEX for which you want to obtain last data collection information and choose Reports => Last Data Collection.

2. In the Last Data Collection popup, enter a number of days (0 through 999).
Advisor determines whether data collection has occurred for each database in the IMSPLEX within this specified number of days. If data collection has not occurred, the database is included in the Last Data Collection report. If data collection has occurred, the database is omitted from the report. To include all databases in the report, enter 0.

3 Click OK.

Figure 102: Last Data Collection report

For more information about the report, see the online Help.

Requesting batch reports for an environment

Database Advisor can generate batch reports that provide the same information that you can view in table format in the IMS Database Management Console.

These reports provide a high-level, consolidated view of all databases in your environment. You can visually compare data for multiple databases and export the data for analysis by a third-party application.

The following batch reports are available:

- Database History
- Dataset History
- Dataset Space
- Area Performance
- Area Space
- Database Summary

You can generate a batch report through the following techniques:

- Enter a request through the IMS Database Management Console
Submit a batch job to execute the SUMRYRPT command of the CPCBATCH utility.

**Note**
Generation of batch environment reports requires a license for the MAXM Database Advisor product.

---

### Requesting batch reports from the console

When you request batch reports from the console, Database Advisor processes each request in the background and saves the data in a batch report file on the mainframe system.

You can view the report on the console (unless the report exceeds 4000 lines) or from your mainframe system interface (such as ISPF).

**To request and view a batch report from the console**

1. In the Navigation window, right-click an IMSPLEX object, and choose **Reports => Batch Reports => Request Batch Reports**.

2. Provide the requested information in the Report Options dialog box, and click **OK**.

3. Wait for message **BMCMXA340682I** in the console’s Messages window.

   This message contains the data set name of the batch report file. The console retains this name until termination of your current console session.

4. View the generated report through the console or in ISPF:

   - In the Navigation window of the console, right-click the IMSPLEX object, and choose **Reports => Batch Reports => View Existing Batch Report**. Provide the name of the batch report file to view. (If the file was generated during the current console session, you can select the file from the drop list.)

   - In ISPF View or Edit, provide the name of the batch report file to be displayed.

---

### Generating batch reports with the SUMRYRPT command

You can use the SUMRYRPT command of the CPCBATCH utility to request a batch report.

The utility writes the report to a member of a specified partitioned data set, from which you can view and analyze the data.
To generate a batch report with the SUMRYRPT command

1 Create JCL to run the SUMRYRPT command of the CPCBATCH utility.
   For more information, see “JCL for the SUMRYRPT command” on page 446.

2 Specify control statements for the SUMRYRPT command.
   These control statements identify the inputs and outputs for the process, and
   specify the options to use. For more information, see “Control Statements for the
   SUMRYRPT command” on page 447.

3 Submit the JCL for execution.

4 Check the results of the job.
   For more information, see “Return codes for the SUMRYRPT command” on page
   453.

5 View the generated report through the console or in ISPF:
   - In the Navigation window of the console, right-click the IMSPLEX object, and
     choose Reports => Batch Reports => View Existing Batch Report. Provide
     the name of the batch report file to view. (If the file was generated during the
     current console session, you can select the from the drop list.)
   - In ISPF View or Edit, provide the name of the batch report file to be displayed.

JCL for the SUMRYRPT command

The following figure shows example JCL to execute the SUMRYRPT command of the
CPCBATCH utility.

Table 56 on page 447 describes the valid statements in this JCL.

Figure 103: JCL for the SUMRYRPT command

```
//CPCBATCH JOB (acct),'BATCH REPORTS',MSGCLASS=X,REGION=OM,
//CLASS=A,NOTIFY=&SYSUID
//CPCBAT01 EXEC  PGM=CPCBATCH,REGION=OM
//STEPLIB  DD DISP=SHR,DSN=BMC.OPTIONS.GLOBAL   < BMC GLOBAL OPTIONS
//         DD DISP=SHR,DSN=BMCDLI.LOAD          < BMC DLI LOAD LIBRARY
//         DD DISP=SHR,DSN=BMCCPC.LOAD          < BMC CPC LOAD LIBRARY
//outfile DD DISP=SHR,DSN=outfileDataSetName < optional
//CPCSYSIN DD  *
SUMRYRPT control statements
/*
```
Table 56: JCL for the SUMRYRPT command

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB</td>
<td><em>(Required)</em> Specify your standard JOB statement.</td>
</tr>
<tr>
<td>EXEC</td>
<td><em>(Required)</em> Specify CPCBATCH as the program to be executed.</td>
</tr>
<tr>
<td>STEPLIB DD</td>
<td><em>(Required)</em> Include the library or libraries that contain your BMC</td>
</tr>
<tr>
<td></td>
<td>global options modules and the MAXM Database Advisor and Cross-Product</td>
</tr>
<tr>
<td></td>
<td>Connectivity (CPC) executable load modules.</td>
</tr>
<tr>
<td>outfile DD</td>
<td><em>(Optional)</em> Identify the partitioned data set to contain the batch report.</td>
</tr>
<tr>
<td></td>
<td>The ddname that you specify for this statement is the value of the</td>
</tr>
<tr>
<td></td>
<td>OUTFILE keyword in the control statement data set. If you omit this</td>
</tr>
<tr>
<td></td>
<td>statement, the OUTDATASET keyword is required.</td>
</tr>
<tr>
<td>CPCSYN DD</td>
<td><em>(Required)</em> Identify the data set that contains the SUMRYRPT</td>
</tr>
<tr>
<td></td>
<td>control statements. Specify CPCSYN DD * to include the control</td>
</tr>
<tr>
<td></td>
<td>statements in the JCL.</td>
</tr>
</tbody>
</table>

Control Statements for the SUMRYRPT command

The CPCSYN control statement data set contains 80-character fixed-length records that control the actions of the CPCBATCH utility.

For general information about how to specify control statements, see “Control Statements for the EXCEPLST command” on page 312.

Figure 104 on page 447 shows all available keywords for the SUMRYRPT command.

Figure 104: Control statements for the SUMRYRPT command

```c
//CPCSYN DD *
SUMRYRPT -
REPORT_TYPE(report) -
OUTDS(outputDataSetName) or ODS(outputDataSetName) -
/* OUTFILE(ddname) or OFILE(ddname) - do not specify with OUTDS */ -
RECON_DSN(reconDataSetName) -
SORT_COLUMN(columnName) -
DETAIL_LEVEL(detailLevel) -
READ_DBINFO(LAST or ALL) -
*/
```

Table 57 on page 448 describes the valid keywords for the SUMRYRPT command.
### Table 57: Keywords for the SUMRYRPT command

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Valid Values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT_TYPE</td>
<td>▪ DBHIST ▪ DSHIST ▪ DSSPACE ▪ AREAPERF ▪ AREASPC</td>
<td>None</td>
<td><em>(Required)</em> Specify one of the following reports to produce:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ DBHIST—Database History</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ DSHIST—Dataset History</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ DSSPACE—Dataset Space</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ AREAPERF—Area Performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ AREASPC—Area Space</td>
</tr>
<tr>
<td>RECON_DSN</td>
<td>Data set name (maximum 44 characters)</td>
<td>None</td>
<td><em>(Optional)</em> Specify the data set name of a RECON data set to read. If you</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>omit this keyword, the CPCBATCH utility searches all RECONs that are</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>identified in the IMS environment data in the repositories.</td>
</tr>
<tr>
<td>OUTDATASET or</td>
<td>Data set name (maximum 44 characters)</td>
<td>None</td>
<td><em>(Required if OUTFILE is not specified)</em> Specify the data set name of the</td>
</tr>
<tr>
<td>OUTHLQ or ODS or</td>
<td></td>
<td></td>
<td>output data set to contain the generated report. It can be a pre-existing</td>
</tr>
<tr>
<td>OHLQ</td>
<td></td>
<td></td>
<td>data set, or the CPCBATCH utility can allocate it dynamically. Do not</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>specify OUTDATASET if you specify OUTFILE.</td>
</tr>
<tr>
<td>OUTFILE or OFILE</td>
<td>DDname (maximum 8 characters)</td>
<td>None</td>
<td><em>(Required if OUTFILE is not specified)</em> Specify the ddname of the output</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>data set to contain the generated report. Do not specify OUTFILE if you</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>specify OUTDATASET.</td>
</tr>
<tr>
<td>SORT_COLUMN</td>
<td>Column name (see <em>Valid values for the SORT_COLUMN</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>keyword on page 449)</td>
<td>DBDNAME</td>
<td><em>(Optional)</em> Specify the name of the column to use as the sort field for</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the data. Use an underscore character instead of a space between words.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>For example, to sort the data on the Count of Segments field, specify</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COUNT_OF_SEGMENTS.</td>
</tr>
<tr>
<td>DETAIL_LEVEL</td>
<td>▪ D ▪ M ▪ Y</td>
<td>D</td>
<td><em>(Optional)</em> Specify the type of repository records to process:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ D—Detail</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ M—Monthly</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>▪ Y—Yearly</td>
</tr>
<tr>
<td>READ_DBINFO</td>
<td>▪ LAST ▪ ALL</td>
<td>LAST</td>
<td><em>(Optional)</em> Specify whether to use only the most recent record for each</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>database or use all records.</td>
</tr>
</tbody>
</table>
Valid values for the SORT_COLUMN keyword

The following table lists the valid values and sort orders for the SORT_COLUMN keyword, according to report.

<table>
<thead>
<tr>
<th>Report</th>
<th>Column name</th>
<th>Sort order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database History</td>
<td>DBDNAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>PARTITION_NAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RUN_DATE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RUN_TIME</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>DATABASE_ORG</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DB_RECORDS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>COUNT_OF_SEGMENTS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>COUNT_OF_SEGMENTS_IN_OVFLW</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>COUNT_OF_SPLIT_SEGMENTS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>SPLIT_TO_DIFF_BLOCK</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>IO_PROBABILITY</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>DB_RECORD_SIZE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>PERCENT_RAA_FREE_SPACE</td>
<td>Ascending</td>
</tr>
<tr>
<td>Report</td>
<td>Column name</td>
<td>Sort order</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Dataset History</td>
<td>DBDNAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>PARTITION_NAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DSGID</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RUN_DATE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RUN_TIME</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>TOTAL_NO_BLOCKS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>TOTAL_NO_BITMAP_BLOCKS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLOCKS_FULL</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLOCKS_PART_FULL</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLOCKS_EMPTY</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLOCKS_UNUSED</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BLOCKS_OF_FREESPACE</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>TOTAL_BYTES_OF_SPACE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_SEG_PREFIX</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_SEG_DATA</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_SEG_PAD</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_FREESP_USABLE</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_FREESP_UNUSABLE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>bytes_of_slack</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_DLI_OVERHEAD</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BYTES_VSAM_CI_OVERHEAD</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BLOCKS_WO_SPACE_BITMAP</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_FSE_HOLD_LARGE_SEG</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_FSE_NOHOLD_SMALL_SEG</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLKS_WITH_WARN_MSGS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_BLKS_WITH_SLACK_BYTES</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BLKS_ADD_POST_LOAD_RELOAD</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>PERCENT_USABLE_FREESPACE</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>BYTES_OF_FREESP_USABLE_LS</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>PERCENT_USABLE_FREESPACE_LS</td>
<td>Ascending</td>
</tr>
<tr>
<td>Report</td>
<td>Column name</td>
<td>Sort order</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Dataset Space</td>
<td>DBDNAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>PARTITION_NAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DSGID</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RUN_DATE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RUN_TIME</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_VOLUMES</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO_EXTENTS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>HIGH_USED_RBA</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BLOCKS_ALLOCATED</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>BLOCKS_USED</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>CI_SPLITS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>CA_SPLITS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>NO SEGMENTS</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>AVAIL_EXTENTS</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>AVAIL_VOLUMES</td>
<td>Ascending</td>
</tr>
<tr>
<td>Report</td>
<td>Column name</td>
<td>Sort order</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Area Performance</td>
<td>DBDNAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>AREANAME</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RUN_DATE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RUN_TIME</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>SEGMENT_DDEP_COUNT</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>SEGMENT_SDEP_COUNT</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>SYNONYM_CHAIN_LENGTH</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>UOWS_USING_DOVF</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>UOWS_USING_IOVF</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RECORD_COUNT</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RECORD_SIZE</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RECORD_COUNT_DOVF</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RECORD_COUNT_IOVF</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RECORD_IO</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>SYNONYM_CHAIN_IO</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>FRAGMENTATION</td>
<td>Descending</td>
</tr>
<tr>
<td></td>
<td>RAP_FREE_TOTAL_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RAP_FREE_USABLE_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>RAP_FREE_BLOCKS_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DOVF_FREE_TOTAL_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DOVF_FREE_USABLE_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>DOVF_FREE_BLOCKS_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>IOVF_FREE_TOTAL_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>IOVF_FREE_USABLE_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>IOVF_FREE_BLOCKS_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>SDEP_FREE_TOTAL_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>SDEP_FREE_USABLE_PCT</td>
<td>Ascending</td>
</tr>
<tr>
<td></td>
<td>SDEP_FREE_BLOCKS_PCT</td>
<td>Ascending</td>
</tr>
</tbody>
</table>
Return codes for the SUMRYRPT command

The following table explains the return codes that the SUMRYRPT command can issue.

Table 58: Return codes for the SUMRYRPT command

<table>
<thead>
<tr>
<th>Code</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The function completed successfully.</td>
</tr>
<tr>
<td>8</td>
<td>A CPC connection error occurred.</td>
</tr>
<tr>
<td>12</td>
<td>No output data set was defined.</td>
</tr>
</tbody>
</table>
### Generating a Threshold Settings batch report

You can generate a report showing the threshold settings you have defined. The Threshold Settings batch report is a list of the threshold settings, as selected through specified filters. You use the DLIATRPT utility to request the report. The utility writes the report to an output DD data set, from which you can view and analyze the data.

#### To generate the Threshold Settings batch report

1. Create JCL to run the DLIATRPT utility.
   
   For more information, see “JCL for the DLIATRPT utility” on page 454.

2. Specify control statements.
   
   These control statements identify the inputs and outputs of the process, and options and filters to use. For more information, see “Control statements for the DLIATRPT utility” on page 455.

3. Submit the JCL for execution.

4. View the output.
   
   For more information, see “Example Threshold Settings reports” on page 457.

#### JCL for the DLIATRPT utility

The following figure shows example JCL to execute the DLIATRPT utility.

Table 34 on page 312 describes the valid statements in this JCL.

**Figure 105: Sample JCL for the DLIATRPT utility**

```jcl
THRESPT JOB (acct), 'THRESHOLD LIST', MSGCLASS=X, REGION=0M,
   CLASS=A, NOTIFY=&SYSUID
  ********************************************************************
   *  CPC THRESHOLD REPORT  
   ********************************************************************
   **  CPC THRESHOLD REPORT  
    **                     
   /*  CPC THRESHOLD REPORT  
   
   THRTPT EXEC PGM=DLIATRPT
   STEPLIB DD DISP=SHR, DSN=BMC.OPTIONS.GLOBAL
   
   SYSOUT DD DISP=SHR, DSN=PRODUCT.LOAD
   
   /*  CPC THRESHOLD REPORT  
   
   /*  CPC THRESHOLD REPORT  
   */
   SELECT (DBD(HDD))
   REPORT (ALL)
   */
```
Table 59: JCL for the DLIATRPT utility

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB</td>
<td>(Required) Specify your standard JOB statement.</td>
</tr>
<tr>
<td>EXEC</td>
<td>(Required) Specify DLIATRPT as the program to be executed.</td>
</tr>
<tr>
<td>STEPLIB DD</td>
<td>(Required) Include the library or libraries that contain your BMC global options modules and the MAXM Database Advisor and Cross-Product Connectivity (CPC) executable load modules.</td>
</tr>
<tr>
<td>SYSOUT DD</td>
<td>(Optional) Identify the data set to contain the report. The output data set must have a logical record length (LRECL) of 133.</td>
</tr>
<tr>
<td>SYSIN DD</td>
<td>(Required) Identify the data set that contains the DLIATRPT control statements. Specify SYSIN DD * to include the control statements in the JCL.</td>
</tr>
</tbody>
</table>

Control statements for the DLIATRPT utility

The DLIATRPT control statement data set contains 80-character, fixed-length records that control the actions of the DLIATRPT utility.

SYSIN syntax

In the SYSIN data set, you can specify control statements by using the following syntax elements:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyword</td>
<td>Keywords invoke options. All keywords are nonpositional. You specify a keyword with a value following it in parentheses. Keywords are recognized by any prefix sufficient to identify them. For example: REPORT(ALL), REP(AL), and R(A) are all valid and equivalent.</td>
</tr>
<tr>
<td>Value</td>
<td>Values are recognized by any prefix sufficient to identify them. For example: REPORT(ALL), REP(AL), and R(A) are all valid and equivalent.</td>
</tr>
<tr>
<td>Separator</td>
<td>When you require a separator, use a comma.</td>
</tr>
<tr>
<td>Continuation characters</td>
<td>Use the plus sign (+) to continue control statements that do not fit on a single line of input. The continuation character must be the last nonblank character.</td>
</tr>
</tbody>
</table>

Keywords for the DLIATRPT utility

The following table describes the keywords for the DLIATRPT utility. All keywords are optional. They default as indicated in the table.
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Valid Values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT</td>
<td>ALL</td>
<td>ALL</td>
<td>Determines whether to include all effective thresholds or just explicit thresholds</td>
</tr>
<tr>
<td></td>
<td>EXPLICIT</td>
<td></td>
<td>- ALL reports all effective values for thresholds, including their inheritance source.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- EXPLICIT generates a tiered report of only the explicit thresholds set at specific levels.</td>
</tr>
<tr>
<td>SELECT</td>
<td>RECON</td>
<td>All objects are included</td>
<td>Limits the report to specific object types</td>
</tr>
<tr>
<td></td>
<td>DBGROUP</td>
<td></td>
<td>Any number of SELECT statements can be specified.</td>
</tr>
<tr>
<td></td>
<td>DBD</td>
<td></td>
<td>Objects are included if they match all qualifiers of one of the SELECT statements. Omitted qualifiers are ignored.</td>
</tr>
<tr>
<td></td>
<td>PARTITION</td>
<td></td>
<td>Each value is formatted like a keyword followed by the value in parentheses. Multiple values can be specified in one SELECT statement. Separate values with a comma.</td>
</tr>
<tr>
<td></td>
<td>DSGNUM</td>
<td></td>
<td>Examples:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SELECT(RECON(HLQ.IMSA.RECON1))</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SELECT(RECON(HLQ.IMSA.RECON1),DBGROUP(SMALLDBS),DBD(SMDB1))</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SELECT(DBD(SMDB1))</td>
</tr>
<tr>
<td>INCLUDE</td>
<td>thresholdName</td>
<td>All thresholds are included</td>
<td>Includes all thresholds by default</td>
</tr>
<tr>
<td>EXCLUDE</td>
<td></td>
<td></td>
<td>- If any thresholds are specified with the INCLUDE statement, only those thresholds are included in the report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- If any thresholds are specified with the EXCLUDE statement, all thresholds will be shown except for those specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>As a result, it does not make much sense to have both INCLUDE and EXCLUDE statements in the same job.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Just like values, threshold names are recognized by their shortest unique prefix.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Most threshold names contain multiple words. Reports use underscores to separate words in threshold names. INCLUDE and EXCLUDE statements should also specify them this way. For example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>INCLUDE(EXTENTS_EXCEED_OSAM)</td>
</tr>
<tr>
<td>Keyword</td>
<td>Valid Values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SOURCE</td>
<td>YES NO</td>
<td>YES</td>
<td>Determines whether to include the inheritance source for every threshold</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ YES includes the inheritance source for every threshold in the report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ NO does not include the inheritance source for every threshold in the report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong> REPORT(ALL) includes source by default. REPORT(EXPLICIT) excludes source by default.</td>
</tr>
<tr>
<td>VERBOSE</td>
<td>YES NO</td>
<td>NO</td>
<td>Determines whether to display objects that do not have explicit thresholds in the report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ YES displays objects that do not have explicit thresholds.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In the report, objects with no thresholds will show the line <strong>&lt;&lt;NO THRESHOLDS SET AT THIS LEVEL&gt;&gt;</strong> instead of threshold values.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ NO does not display objects that do not have explicit thresholds.</td>
</tr>
</tbody>
</table>

**Example Threshold Settings reports**

The following figures show examples of the Threshold Settings report.
### Example

#### Figure 106: Example using REPORT(ALL)

<table>
<thead>
<tr>
<th>THRESHOLD</th>
<th>VALUE</th>
<th>SOURCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMSPLEX :JXWPL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RECON DSN :USER1.MVSJWX1.JXWD.RECON1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB GROUP :HD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alloc failed range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Alloc failed</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Alloc freespace(ls) %</td>
<td>0</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Alloc freespace %</td>
<td>0</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Detect area allocate failure</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area allocate failure level</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area io error</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area stopped range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area stopped</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area unavailable</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area used</td>
<td>0</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Area waiting for migration</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Database availability</td>
<td>5</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Send immediate e-mail</td>
<td>Y</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Max casplits</td>
<td>0</td>
<td>DBGROUP</td>
</tr>
<tr>
<td>Ca splits %</td>
<td>20</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Max cisplits</td>
<td>0</td>
<td>DBGROUP</td>
</tr>
<tr>
<td>Ci splits %</td>
<td>20</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Consolidate e-mail messages</td>
<td>N</td>
<td>DBGROUP</td>
</tr>
<tr>
<td>Dbd area init fail range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd/area init fail</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd area locked range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd/area locked</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd area not open range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd/area not open</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd area stopped range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd/area stopped</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd cmd pending range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd cmd pending</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd deleted by olchng range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd deleted by olchng</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd freespace low range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd freespace low</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd in error range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbd in error</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbr cmd pending range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbr cmd pending</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbrc auth fail range</td>
<td>00-24</td>
<td>SYSPLEX</td>
</tr>
<tr>
<td>Dbrc auth fail</td>
<td>N</td>
<td>SYSPLEX</td>
</tr>
</tbody>
</table>
Example

Figure 107: Example using REPORT(EXPLICIT)

<table>
<thead>
<tr>
<th>IMSPLEX</th>
<th>JXWPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON DSN</td>
<td>USER1.MVSJXW1.JXWD.RECON1</td>
</tr>
<tr>
<td>DB GROUP</td>
<td>HD</td>
</tr>
</tbody>
</table>

THRESHOLD                        VALUE
------------------------------   --------
Max casplits                            0
Max cisplits                            0
Warning days - data set size           31
Warning days - extents + volum         32
Warning days - free space              32
Warning days - database perfor         33
Warning days - ci + ca splits          33
Warning days                            31

IMSPLEX     :JXWPL          
RECON DSN   :USER1.MVSJXW1.JXWD.RECON1 
DB GROUP    :HD             
DATABASE    :HDD            
D.B. DATASET:00001         
THRESHOLD                        VALUE
------------------------------   --------
Max casplits                            0
Max cisplits                            0
Ci splits %                             35
Consolidate e-mail messages             N
I/o probability growth %               93
Warning days - data set size            8
Warning days - extents + volum          5
Warning days - free space              34
Warning days - database perfor         42
Warning days - ci + ca splits          29
Warning days                            7
Warning e-mail address 1
USER1@bmc.com

Analyzing historical data

The Historical Analysis Wizard guides you through the process of analyzing Advisor historical data for statistics of interest.
These statistics are usually database statistics that have a rate of change above a certain threshold. You use the wizard to select parameters for the analysis. The wizard runs the analysis in the background and notifies you when the analysis is complete. You can then view the results of the analysis in the Workflow Status window.

**Note**

To access and use the Historical Analysis Wizard, you must have a license for MAXM Database Advisor for IMS.

The following procedure outlines the process of using the wizard. For complete information, see the online Help.

**To use the Historical Analysis Wizard**

1. In the Navigation window, right-click the IMSPLEX (RECON) object that contains the data of interest, and select **Historical Analysis Wizard**.

2. On the Select Databases for Analysis panel, specify which databases (and partitions, if applicable) to analyze, and specify the starting date of the analysis.

   You can specify specific or generic values for database and partition names. Generic values contain masking (wildcard) characters.

3. On the Select Data Points panel, select the type of historical data to analyze.

   This selection determines which type of data records that the background process will request from the repository. It also determines the units for one of the growth-rate values on the Select Statistics to Analyze panel.

4. On the Select Statistics to Analyze panel, select individual statistics to analyze.

   You will also use this panel to specify the threshold for the growth rate of the statistics, and specify how far into the future to predict the expected value of the statistics.

5. On the Run Historical Analysis panel, provide information about the mainframe data set that the wizard will use to contain the results of the analysis.

6. On the Summary panel, determine whether the analysis process started successfully.

7. Receive notification that the request has completed.

   The Historical Analysis Wizard attempts to notify you of the final status of the request by displaying a popup dialog that contains a link to the Workflow Status window.
8 View the status of the request in the Workflow Status window.

To access this window, you can:

- Click the link in the notification popup dialog.
- Right-click the IMS Databases (host) object or an IMSPLEX (RECON) object, and select **Workflow Status**.

Historical Analysis Wizard requests are identified by **HANL** in the **Type** field.
Requesting batch reports for an environment
Using database administration tools

This section contains information about using the database administration tools that are provided with BMC database administration products for IMS.

For more information, view the Quick Course "MAXM Database Advisor-Using the DBA Toolkit."

Disassembling IMS control blocks

This section describes how to use the disassembly tool to disassemble selected IMS control blocks.

For more information, view the Quick Course "MAXM Database Advisor-Using the Disassembly Tool."

Overview of the disassembly tool

The disassembly tool offers a quick and convenient way to produce source code for the selected IMS control block.

You can use the disassembly tool to perform the following tasks:

- Rebuild lost source libraries
- Verify the active source version
- Save, change, or reuse source code

The tool disassembles database definition (DBD), program specification block (PSB), application control block (ACB), dynamic allocation (MDA), and message format service (MFS) control blocks. DBDs can be disassembled at IMSPLEX and database
levels. PSB, ACB, MDA, and MFS control blocks can be disassembled only at the
IMSPLEX level.

**Note**
The MFS disassembly function is available only if you are licensed for Database Advisor.
Disassembly of DEDBs from the ACBLIB is not available at this time.

After disassembly you can save the source code to your personal computer or to a
partitioned data set on the mainframe. If you disassembled object code for a DBD, you can also view a DBD map (see “Mapping DBDs” on page 470).

The BMC DATABASE INTEGRITY PLUS product provides extensive tools for working with IMS control blocks, including control block validation, masked searches, library cross-references, comparisons, reports, and audits.

### Disassembling DBD, PSB, ACB, and MDA control blocks

You can disassemble a DBD, PSB, ACB, or MDA control block in the Navigation window.

1. In the Navigation window, right-click the IMSPLEX object (or database object to disassemble a DBD at the database level) and choose **Disassemble => DBD/PSB/ACB/MDA** (or choose **Disassemble** at the database level).
The DBD/PSB/ACB/MDA window (Figure 108 on page 465) is displayed.

**Figure 108: DBD/PSB/ACB/MDA window**

![Image of DBD/PSB/ACB/MDA window]

*Note*

If you selected a database for disassembly of a DBD, the object code is automatically disassembled and displayed on the Source tab; you do not have to perform Step 2 on page 465.

2 *(if required)* On the Source tab, specify the required parameters and click **Disassemble**.

The parameters are specified in the top part of the window, and the source code is displayed in the bottom part of the window. For more information, see the online Help.

**Figure 108 on page 465** shows disassembled object code for a DBD in the DBD/PSB/ACB/MDA window.
Disassembling an MFS control block

To use the MFS disassembly function, you must have a license for Database Advisor.

1. In the Navigation window, right-click the IMS PLEX object and choose Disassemble => MFS.

The MFS window is displayed. This window is similar to the DBD/PSB/ACB/MDA window (Figure 108 on page 465).

2. In the MFS window, specify the required parameters and click Disassemble.

For more information about each parameter, including descriptions, valid values, and suggested defaults, see the online Help.

**Note**

The disassembled MFS code must be modified before the code is used as input to the MFS language utility.

Disassembling control blocks in batch

This section describes how to use the batch disassembly tool to disassemble IMS control blocks.

**JCL requirements**

The following table lists JCL statements for the batch disassembly tool.

<table>
<thead>
<tr>
<th>Table 60: Batch disassembly tool JCL statements</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Statement</strong></td>
</tr>
<tr>
<td>EXEC</td>
</tr>
<tr>
<td>STEPLIB</td>
</tr>
<tr>
<td>CPCSYSIN</td>
</tr>
<tr>
<td>CPCLOG</td>
</tr>
</tbody>
</table>

Figure 109 on page 466 shows example JCL for the batch disassembly tool.

**Figure 109: Batch disassembly tool JCL**

//DISASS1 EXEC PGM=CPCBATCH,TIME=1439
//STEPLIB DD DISP=SHR,DSN=BMC.LOAD
Table 61 on page 467 lists keywords that are used to control processing of the batch disassembly tool.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON_DSN</td>
<td>RECON1 data set name</td>
<td>None</td>
<td>Specify the RECON data set name for the batch disassembly tool.</td>
</tr>
<tr>
<td>SOURCE_DSN</td>
<td>Data set name</td>
<td>None</td>
<td>Specify the data set name of the library containing the source members that are disassembled. The data set can be a DBDLIB, PSBLIB, ACBLIB, or MFSLIB.</td>
</tr>
</tbody>
</table>

**Note:**

- You must specify RECON_DSN or SOURCE_DSN.
- If you specify RECON_DSN and SOURCE_DSN values, members that are disassembled are selected from the specified SOURCE_DSN data set.
- If you omit the SOURCE_DSN value, SOURCE_DSN data set names for the specified TYPE are obtained from the environment information of the IMSPLEX for the specified RECON_DSN.
- If you specify a SOURCE_DSN value and HALDBs are selected for disassembly, the RECON_DSN value must be used to specify the data set name of the RECON containing the HALDBs.
- If you specify a SOURCE_DSN, the specified TYPE must be the same as the type of the source data set.
### Keyword Table

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TYPE</strong></td>
<td></td>
<td>None</td>
<td><em>(conditional)</em> Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td>DBD</td>
<td></td>
<td>(conditional) Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td>PSB</td>
<td></td>
<td>(conditional) Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td>ACB</td>
<td></td>
<td>(conditional) Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td>MDA</td>
<td></td>
<td>(conditional) Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td>FORMAT or FMT</td>
<td></td>
<td>(conditional) Specify the disassembly type. Specify FORMAT or FMT for MFS disassembly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong> To use the MFS disassembly function, you must have a license for Database Advisor.</td>
</tr>
<tr>
<td><strong>DBGROUP</strong></td>
<td>DBD names</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name or name pattern of the DBDs (from the DBGROUP) that are selected for disassembly. This applies for TYPE (DBD) only.</td>
</tr>
<tr>
<td><strong>OBJECT</strong></td>
<td>PDS names</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name or name pattern of PDS members selected for disassembly.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ The OBJECT and DBGROUP keywords are mutually exclusive.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ If you omit both the OBJECT and DBGROUP, all members of source data sets are disassembled.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ The OBJECT can be a name pattern with asterisks(*) as wild card characters. An asterisk as the last character in a pattern treats all remaining characters as wild card characters; however, if trailing spaces are required as part of the pattern, ampersands(&amp;) are used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Example:</strong> SOURCE_DSN (TEST.DBDLIB) has the following members:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ OSAMTSS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ OSAMTSS1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ OSAMTST</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ OSAMTST1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT(OSAMTS*) selects all four members for disassembly OBJECT(OSAMTST*) selects OSAMTST and OSAMTST1 for disassembly</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT(OSAMTS*&amp;) selects OSAMTSS and OSAMTST for disassembly</td>
</tr>
</tbody>
</table>

---

**Disassembling IMS control blocks**

Database Products for IMS Advisors and Toolkit User Guide

468
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUTFILE or OFILE</td>
<td>Output data set ddname</td>
<td>None</td>
<td><em>(conditional)</em> Specify the ddname for the output data set. The OUTFILE keyword is required if OUTDATASET is not specified. OUTFILE is mutually exclusive with OUTDATASET. The output is placed in this data set. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records.</td>
</tr>
<tr>
<td>OUTDATASET or ODS</td>
<td>Output data set name</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name for the output data set. The OUTDATASET keyword is required if OUTFILE is not specified. OUTDATASET is mutually exclusive with OUTFILE. The value specifies the name of an existing data set in which to place the output. The data set is dynamically allocated during processing. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records. If the data set is partitioned, member names can be specified in parentheses following the data set name. The member name value can be any of the values described by the MEMBER keyword.</td>
</tr>
</tbody>
</table>

**Example:** OUTDATASET(my.dbd.source(&DBD)) is equivalent to OUTDATASET(my.dbdsource) MEMBER(&DBD)

| MEMBER              | Output PDS member name                               | DBD name or partition name or change accumulation group name or log subsystem name | Specify the output PDS member name or name pattern. The MEMBER keyword controls the member names of the output data set when the data set is a PDS or PDSE. If output is being generated for multiple databases and the value of MEMBER is a specific name, the output for all databases will be placed in that member. A value of `&DBD` indicates that the DBD name or partition name should be used for the member name. The value can be a pattern for multiple PDS member names. If the last character of the value is an asterisk (*), the preceding characters will be used as a prefix and the characters beginning with the position of the asterisk through the eight characters of the name will be numeric values beginning with zeros. |

**Example:** MEMBER(DBD*) generates member names DBD00000, DBD00001, DBD00002, ...
Mapping DBDs

This section describes how to use the DBD mapping tool. The DBD mapping tool provides a quick and convenient way to look at a graphical representation of the segment structure in a DBD.

For more information, view the Quick Course "MAXM Database Advisor - Using the DBD Mapper."

Overview of the DBD mapping tool

The DBD mapping tool provides an easy way to view, print, and export the following information about the segment structure of a database:

- Segment hierarchy
- Internal logical relationships
- Secondary properties

When you disassemble object code for a DBD, the DBD mapping tool generates a DBD map (Figure 110 on page 471). Disassembling object code for a DBD is a prerequisite for viewing a DBD map.
The DBD Map tab has the following panes:

- The top left pane shows a panning control (a rectangular outline) which you can position to the part of the map that you want to view. For example, to view the top left part of the DBD map, click the top left corner of the panning control.

- The bottom left pane shows a drop-down menu where you can select segments or logical relationships to view.

- The right pane shows the DBD map.

Using the map viewing option buttons under the DBD map (Zoom In, Zoom Out, Print, Export to JPEG), you can zoom in and zoom out, focus on an area of the DBD map, and view internal logical relationships. You can also print and export a DBD map.
Creating and viewing DBD maps

You can create and view a DBD map in the DBD/PSB/ACB/MDA window.

1 Disassemble object code for a DBD by performing the steps in “Disassembling DBD, PSB, ACB, and MDA control blocks” on page 464.

The DBD is disassembled and is displayed in the DBD/PSB/ACB/MDA window (Figure 108 on page 465).

2 In the DBD/PSB/ACB/MDA window, click the DBD Map tab.

The DBD map is displayed.

Navigating a DBD map

You can view segment information by using the following navigation and informational features:

1 Perform one of the following steps:

■ To bring a segment into center view, select the segment that you want to view from the list of segments in the bottom left pane.

■ To view segment properties, place your mouse pointer on a segment in the DBD map.

  Segment properties are displayed in a flyover window.

■ To view internal logical relationships, select **Logicals** from the drop-down list in the bottom left pane and select a segment pair.

  The DBD map shows the relationship by connecting logically related segments with a curved line.

■ To increase or decrease the size of the DBD map, click **Zoom In** or **Zoom Out**.

Using HALDB modeling and migration tools

The interactive high availability large database (HALDB) modeling and migration tools help you migrate traditional full-function IMS databases to HALDBs, and model parameters for managing existing HALDB partitions.
To use the HALDB modeling and migration tools, you must have a license for MAXM Database Advisor.

**Purpose of the tools**

Many organizations are migrating to HALDBs to increase database capacity, improve data availability, and simplify partition management.

After migration, some organizations want to rebalance or split partitions to manage their HALDBs more effectively. These tasks can require extensive analysis of your existing databases and current and future requirements.

This analysis can be difficult and time-consuming, even for experienced DBAs. You must collect current information about your existing databases or partitions and evaluate it according to many parameters. Because some parameters interact with each other, a change in one parameter can require changes in others. You must also consider and accommodate the potential for database growth and change over time.

After determining the correct parameters to use, you must create all of the elements to set up the new or changed HALDB structures. Manual creation of the JCL, control statements, and definitions consumes time and is prone to error. The HALDB modeling and migration tools automate and simplify these tasks.

For example, you can use the Generate Model wizard to propose a set of HALDB parameter values to use for converting an existing full-function database. The tool automatically gathers information about the database, processes the information against your proposed values, and reports the results as a set of recommendations. After viewing the recommendations, you can change your proposed values and request another analysis, using the same or refreshed database information. You can repeat this process as many times as needed to determine the best values for your database.

You can then use the sample JCL, control statements, and DBRC definitions that the tool generated to set up the new HALDB. This automated generation reduces the time and effort required for creating setup elements and ensures that they are free of errors and omissions.

**Supported database types**

The HALDB modeling and migration tools work the following types of databases:

- Databases that are defined with the hierarchical direct access method (HDAM)
- Databases that are defined with the hierarchical indexed direct access method (HIDAM)
- Secondary index databases (INDEX)
- Partitioned HDAM (PHDAM) databases
- Partitioned HIDAM (PHIDAM) databases
- Partitioned INDEX (PSINDEX) databases

**Recommendations from the tools**

Using your proposed specifications, the tools recommend the following items:
- Randomizing parameters and key ranges for a PHDAM database
- Key ranges for a PHIDAM database or PSINDEX database
- IDCAMS delete/define statements
- DBRC control cards
- JCL to make the requested changes

**Getting started with HALDB modeling and migration tools**

HALDB tools use workflows to manage HALDB-related tasks.

During the workflow process, Advisor gathers information and creates the JCL, control statements, and definitions for performing the selected task. You can then access and work with the results in the Maintain Existing Workflow window.

**To get started with HALDB modeling and migration tools**

1. Right-click a valid object and select the tool.

For more information, see the online Help.

The following HALDB modeling and migration tools are available when you right-click the indicated object. These tools accomplish the indicated tasks and work with the indicated type of workflows:
<table>
<thead>
<tr>
<th>Tool</th>
<th>Selected object</th>
<th>Task</th>
<th>Workflow type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain Existing Workflow window</td>
<td>IMSPLEX (RECON); full-function database; HALDB; HALDB partition</td>
<td>View, edit, submit, and delete existing workflows and complete Migrate to HALDB workflows.</td>
<td>All</td>
</tr>
<tr>
<td>HALDB Migration Workflow Definition wizard</td>
<td>IMSPLEX (RECON)</td>
<td>Create a workflow to select a set of full-function databases and define parameters for converting those databases (and their associated databases) to HALDBs.</td>
<td>Migrate to HALDB</td>
</tr>
<tr>
<td>Generate Model wizard</td>
<td>Full-function database</td>
<td>Create a workflow to convert the selected database to a HALDB.</td>
<td>Database Migrate to HALDB</td>
</tr>
<tr>
<td>PSE Validation wizard</td>
<td>Full-function database; HALDB</td>
<td>Create a workflow to model the effects of a partition selection exit (PSE) on the selected database.</td>
<td>PSE Validation</td>
</tr>
<tr>
<td>Rebalance HALDB wizard</td>
<td>HALDB</td>
<td>Create a workflow to model the effects of different parameters on the balance of records between partitions in an existing HALDB.</td>
<td>HALDB Rebalance</td>
</tr>
<tr>
<td>Split Partition wizard</td>
<td>HALDB partition</td>
<td>Create a workflow to model the effects of different parameters on the division of a selected partition into multiple partitions.</td>
<td>Split Partition</td>
</tr>
</tbody>
</table>

To evaluate PSE routines, Advisor uses the name of the PSE module and the DBRC statements that define the database to determine where the PSE routine would place data from the original database. Advisor then reports record counts and sizes for each partition defined by the DBRC statements.

2 Edit the information and click **Submit**.

The tool creates the JCL and submits it:

1 An EXTRACT job collects segment information from the original database.
2 A SORT job sorts the extracted information.

3 An ESTIMATE job creates recommendations.

4 Advisor writes the recommendations to a PDS data set.
   That data set contains DBRC control statements to update DBRC, IDCAMS
   delete/define statements for the database data sets, and JCL to implement the
   recommendations. You can edit the PDS data set members.

5 When the submitted job completes, a BMCMXA340683I message is displayed in
   the console.

Where to go from here

To view a data set with a summary of the results, from the Maintain Existing
Workflow window, click View.

Using the batch partition and randomizer tuning tool

This section describes how to use the batch partition and randomizer tuning tool.

Note

To use this tool, you must have a license for MAXM Database Advisor.

Overview of the partition and randomizer tuning tool

The partition and randomizer tuning tool is a batch utility that selects hierarchical
indexed direct access method (HIDAM) key ranges or hierarchical direct access
method (HDAM) parameters.

The tool can help you optimize storage of data in IMS full-function databases. It can
also help you to convert a full-function database into a high availability large
database (HALDB).

You can use this tool to obtain randomizing parameters for HDAM databases. The
tool also recommends HALDB partition boundaries and optimal ranges for your
record keys so that data is redistributed evenly across partitions.
Coding JCL

To use the partition and randomizer tuning tool

1 Run the EXTRACT job.

Segment information is collected from the original database.

2 Run the SORT job.

The information that was extracted is sorted.

3 Run the ESTIMATE job.

Control statements that will be used to convert the original database to a HALDB are generated.

Note

Sample JCL for EXTRACT, SORT, and ESTIMATE jobs is stored in the DBUPHAL member of the database utilities (DBU) sample library.

EXTRACT job

The partition and randomizer tuning tool uses segment information to determine appropriate partition boundaries for new HALDBs. The segment information is contained in the EXTRACTO data set that is generated by the DLIAANL2 program.

Figure 111 on page 477 shows JCL for generating the EXTRACTO data set.

Figure 111: EXTRACT JCL

000006 //IMS DD DISP=SHR, DSN=IMS.DBDDLIB
000007 //RECON1 DD DISP=SHR, DSN=YOURDBRC.RECON1
000008 //RECON2 DD DISP=SHR, DSN=YOURDBRC.RECON2
000009 //RECON3 DD DISP=SHR, DSN=YOURDBRC.RECON3
000010 //BMCMSG DD SYSOUT=* 
000011 //ANLPRINT DD SYSOUT=* 
000012 //SYSUDUMP DD SYSOUT=* 
000013 //YOURDBD DD DISP=SHR, DSN=YOUR.DATABASE.DATASETS
000014 //EXTRACTO DD DISP=(,CATLG,DELETE),SPACE=(CYL,(99,99),RLSE),
000015 // DCB=(BUFNO=30),UNIT=SYSDA,DSN=YOUR.EXTRACTO.DSN
000016 //EXTRACTI DD DISP=(,CATLG,DELETE),SPACE=(TRK,(1,1)),
000017 // DCB=(LRECL=80,Blksize=3120,RECFM=FB),
000018 // UNIT=SYSDA,DSN=YOUR.EXTRACTI.DSN
000019 //PLUSIN DD *
000020 ANALYZE DBD(YOURBD)
**SORT job**

Input records in the EXTRACTO data set must be sorted. When you are converting to a PHDAM database, a SORT job is always required because partition boundaries are based on key range.

When you are converting to a PHIDAM database, a SORT job is required if the following conditions are true:

- The original database is a HDAM database.
- You are combining multiple HIDAM databases that have overlapping key values.

Figure 112 on page 478 shows JCL for sorting input records.

---

**Figure 112: SORT JCL**

000019 //SORT EXEC PGM=SORT,REGION=4096K  
000020 //SYSOUT DD SYSOUT=*  
000021 //SORTIN DD DISP=SHR,DSN=YOUR.EXTRACTO.DSN  
000022 //SORTOUT DD DSN=YOUR.SORTED.DSN, SPACE=(CYL,(99,99),RLSE),DISP=(,CATLG),UNIT=(DEVXXX),  
000023 //SORTPARM DD *  
000024 //SYSIN DD *  
000025 SORT FIELDS=(9,2,10,17,KL,A),FORMAT=BI <--- KL= DBD ROOT KEY LENGTH  
000026 //SORTWK01 DD UNIT=WORK,SPACE=(CYL,(50,19))  
000027 //SORTWK02 DD UNIT=WORK,SPACE=(CYL,(50,19))  
000028 //SORTWK03 DD UNIT=WORK,SPACE=(CYL,(50,19))

To obtain the key length value, reference the root segment information in the DBD source. Databases that do not have root keys cannot be partitioned.

**Note**

To merge multiple identical original databases into one HALDB, you can concatenate EXTRACTO data sets. This feature supports conversion or reconfiguration of a database that is already partitioned, regardless of the type of partitioning.

---

**ESTIMATE job**

The ESTIMATE job uses the original DBD, the EXTRACTO data set, the EXTRACTI data set, and keyword parameters to determine partition boundaries.

For more information about the keywords, see “Coding the ESTIMATE command and keywords” on page 480.

Figure 113 on page 478 shows JCL for running the partitioning tool batch utility.

---

**Figure 113: ESTIMATE JCL**

//ESTIMATE EXEC PGM=DLIAPHAL  
//--STEPLIB DD DISP=SHR,DSN=YOURDBU.LOAD
Coding the ANALYZE command and keywords

Use the ANALYZE command to run the extract tool batch utility.

Table 62 on page 479 lists keywords that are valid for the ANALYZE command.

Table 62: Extract tool batch utility keywords

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBD</td>
<td>databaseName</td>
<td>None</td>
<td>Specify the name of the database from which to extract segment data.</td>
</tr>
<tr>
<td>PART</td>
<td>partitionName</td>
<td>ALL</td>
<td>Specify the name of the partition from which to extract segment data. Specify ALL to extract segment data from all partitions in the database. Mutually exclusive with the PARTSET and PARTLIST keywords.</td>
</tr>
<tr>
<td>PARTSET</td>
<td>(partitionName, nnnn)</td>
<td>None</td>
<td>Specify the name of a partition to start a range of partitions from which to extract segment data and the number of partitions (nnnn) in the range following that partition. Mutually exclusive with the PART and PARTLIST keywords.</td>
</tr>
<tr>
<td>PARTLIST</td>
<td>(partitionName, partitionName,...)</td>
<td>None</td>
<td>Specify a list of names of partitions from which to extract segment data. The partitions must be adjacent in the database. Mutually exclusive with the PART and PARTSET keywords.</td>
</tr>
</tbody>
</table>
**Coding the ESTIMATE command and keywords**

Use the ESTIMATE command to run the partitioning tool batch utility.

Table 62 on page 479 lists keywords that are valid for the ESTIMATE command.

**Table 63: Partitioning tool batch utility keywords**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>PHDAM, PHIDAM, PSINDEX, HDAM</td>
<td>Input database type</td>
<td>Specify the type of estimation that you want to perform.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------</td>
<td>---------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>COUNT</td>
<td>1-1001</td>
<td>None</td>
<td>Specify the number of partitions to create. If the COUNT keyword is specified, partition boundaries are determined after all input records are read. The total number of database records is divided by the COUNT value to determine the number of records in each partition, and partition boundaries are established within the input records. Mutually exclusive with the RECORDS, PARTSEL, and KEYSIZE keywords.</td>
</tr>
<tr>
<td>SIZE</td>
<td>1-4000</td>
<td>2000 (approximately 2 GB)</td>
<td>Specify the size of a partition in multiples of 1 MB. The SIZE and COUNT keywords are mutually exclusive. If neither keyword is used, the SIZE default is SIZE(2000). If both keywords are specified, the SIZE keyword is used and the COUNT keyword is ignored. If the SIZE keyword is specified, records accumulate until the threshold is reached. The root key is saved at each threshold, and values for HDAM RAPs and blocks are calculated. Segment sizes are adjusted to accommodate new extended pointers. The free space requirement that is specified in the DBD is honored. For HDAM databases, the number of blocks in the RAA is increased to honor the HDAMGROW or the DBGROW value. A unique set of HDAM RAPs and blocks is calculated for each partition. Mutually exclusive with the RECORDS and PARTSEL keywords.</td>
</tr>
<tr>
<td>RECORDS</td>
<td>1-2147483647</td>
<td>None</td>
<td>Specify the number of database records to place in each partition. Mutually exclusive with the SIZE, COUNT, PARTSEL, and KEYSIZE keywords.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>KEYSIZE</td>
<td>1 - [keylength minus 1]</td>
<td>Keylength</td>
<td>Split a single partition into &quot;count&quot; partitions, which are based on a partial root key. This action might be useful if the key contains specific entities (date, state, ALL type, and so on) and you want to split the database into independent partitions on this key component.</td>
</tr>
<tr>
<td>MERGEDSN</td>
<td>Y, N</td>
<td>N</td>
<td>Specify whether to merge any multiple data set groups to create a HALDB that has a single data set group (Y); all input data is accumulated as if no secondary data set groups ever existed; or to preserve the data set group format (N); only the data from the primary data set group is accumulated, and all other data set group data is ignored.</td>
</tr>
<tr>
<td>HDAMGROW</td>
<td>10-50</td>
<td>20</td>
<td>Specify the growth percentage for the PHDAM/HDAM RAA. <strong>Example:</strong> Specify HDAMGROW(20) to leave 20 percent of the RAA empty.</td>
</tr>
<tr>
<td>LASTFREE</td>
<td>Y, N</td>
<td>N</td>
<td>Specify whether or not to leave the last PHIDAM partition empty for future growth (Y) or empty (N). The LASTFREE keyword is valid only for PHIDAM databases when the COUNT or RECORDS keyword is specified (for example, if COUNT(10) and LASTFREE(Y) is specified, the total record size is divided into nine equal partitions and one partition that contains only a few records). If the SIZE keyword is specified, the LASTFREE keyword is ignored because the last partition is already smaller than the other partitions.</td>
</tr>
<tr>
<td>PARTMASK</td>
<td>(namePattern, sequenceType, sequenceStart, sequenceIncrement)</td>
<td>namePattern = first six characters of DBD name sequenceType = alphanumeric sequenceStart = first characters of sequence type sequenceIncrement = 1</td>
<td>Specify the partition name prefix. The PARTMASK keyword is used by DBRC to create the partition ddnames. You can specify a maximum of seven alphanumeric characters. IMS uses the eighth for the data set group. For more information, see Coding the PARTMASK keyword on page 484.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>DSNPREFIX</td>
<td>1-32 alphanumeric characters</td>
<td>MODIFY.MYNAME</td>
<td>Specify the data set name. IMS appends the HALDB partition suffix to the specified data set name.</td>
</tr>
<tr>
<td>NOPINIT</td>
<td>Y, N</td>
<td>N</td>
<td>Specify whether a CHANGE.DB PART(partName) NOPINIT command follows each INIT.PART or CHANGE.PART PART specification in the DBRCOUT data set.</td>
</tr>
<tr>
<td>PARTSEL</td>
<td>PSEModuleName</td>
<td>None</td>
<td>Specify the name of a PSE module to test and determine how it performs with the data from the database. The module must be available in the STEPLIB concatenation of the ESTIMATE step. You must provide a data set with ddname BMCRECON containing the DBRC control statements to define all the database partitions. The utility uses the partition names, DSNPREFIX value, and KEYSTRING values from the BMCRECON data set. It passes the key of each database record to the PSE to determine the partition in which it belongs and reports the sizes and counts for each partition.</td>
</tr>
<tr>
<td>REUSPRTS</td>
<td>Y, N</td>
<td>N</td>
<td>Specify whether to create a DBRC CHANGE.PART statement instead of DELETE.PART and INIT.PART statements when the PARTMASK option generates a partition name that matches a partition name in the input database. Also specifies whether to use the existing partition number to create the data set names in the IDCAMS delete and define statements.</td>
</tr>
</tbody>
</table>
### Coding the PARTMASK keyword

This topic describes how to code the PARTMASK keyword for the ESTIMATE command.

A valid PARTMASK keyword value consists of the following parameters:

- **namePattern**
- **sequenceType**
- **sequenceStart**
- **sequenceIncrement**

Defaults exists for all of these parameters. Therefore, you do not have to specify the PARTMASK keyword. If you do not specify the PARTMASK keyword, Advisor constructs the partition name prefix using up to the first six characters of the DBD name, and appends sequence characters. The partition names are unique and consist of up to seven characters. The number of partitions that are modeled determines the number of sequence characters.

If you specify the PARTMASK keyword, Advisor constructs the partition name using the `namePattern` parameter for the name pattern and the sequencing characters determined by the other parameters. The pattern consists of constant and wild-card characters (asterisks, *). Advisor creates unique names for each partition by replacing wild-card characters with sequencing characters. If you do not specify wild-card characters, Advisor appends the sequence characters to the end. If you...
specify wild-card characters, make sure there are enough adjacent characters to hold the sequence characters for the number of partitions modeled.

**Valid values for the sequenceType parameter**

The following values are valid for sequenceType:

- AN (alphanumeric)—A set of 36 characters: A-Z and then 0-9
- N (numeric)—A set of 10 characters: 0-9
- A (alpha)—A set of 26 characters: A-Z

For all sequence types, Advisor increments the sequence character furthest to the right with the next character in the set. When all characters of the set have been used in that position, Advisor increments the character to the left and the right character cycles back to the first character of the set. Numeric sequencing uses normal decimal sequencing.

You must specify enough sequence characters for all partitions being modeled to have unique partition ddnames. Alphanumeric sequencing only requires two sequence characters for all 1,001 partition ddnames to be unique. Alpha sequencing requires three sequence characters for all 1,001, and requires two sequence characters for up to 676 partitions. Numeric sequencing requires four sequence characters for all 1,001.

### Alphanumeric sequencing

<table>
<thead>
<tr>
<th>Partitions</th>
<th>Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-26</td>
<td>xxxxxAA-xxxxxAZ</td>
</tr>
<tr>
<td>27-36</td>
<td>xxxxxA0-xxxxxA9</td>
</tr>
<tr>
<td>37-72</td>
<td>xxxxxBA-xxxxxB9</td>
</tr>
<tr>
<td>901-936</td>
<td>xxxxxZA-xxxxxZ9</td>
</tr>
<tr>
<td>937-972</td>
<td>xxxxx0A-xxxxx09</td>
</tr>
<tr>
<td>973-1001</td>
<td>xxxxx1A-xxxxx12</td>
</tr>
</tbody>
</table>

### Alpha sequencing

<table>
<thead>
<tr>
<th>Partitions</th>
<th>Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-26</td>
<td>xxxxxAAA-xxxxxAAZ</td>
</tr>
<tr>
<td>27-52</td>
<td>xxxxxABA-xxxxxABZ</td>
</tr>
<tr>
<td>651-676</td>
<td>xxxxxAZA-xxxxxAZZ</td>
</tr>
</tbody>
</table>
### Valid values for the `sequenceStart` parameter

For `sequenceStart`, you must specify a member of the character set corresponding to the sequence type you specify:

- If you specify AN for `sequenceType`, you must specify A or AA for `sequenceStart`.
- If you specify N for `sequenceType`, you must specify 0, 00, 000, or 0000 for `sequenceStart`.
- If you specify A for `sequenceType`, you must specify A, AA, or AAA for `sequenceStart`.

If you specify a start sequence other than the first of the set, you must specify enough members of the set following it for the number of partitions being modeled to have unique sequence characters.

### Valid values for the `sequenceIncrement` parameter

The `sequenceIncrement` parameter allows you to specify a gap in the sequencing. The default value, 1, increments to the next member of the sequence set, leaving no gaps.

**Example**

If you specify 2, Advisor skips every other member when sequencing partition names. That is, the sequence would be 00, 02, 04, and so on, or AA, AC, AE, and so on.

---

### Using reports from the partition and randomizer tuning tool

The partition and randomizer tuning tool generates the following reports:

- DBRCOUT DD statement output
- Database Partition Estimate Report
**DBRCOUT DD statement output**

Output from the DBRCOUT DD statement in a partitioning job contains the control statements that specify the recommended partition boundaries for conversion of a database to a HALDB (or the randomizing parameters for a HDAM database).

Partition boundaries are established when you use the control statements to register the database to DBRC.

Figure 114 on page 487 shows output for the DBRCOUT DD statement.

---

**Figure 114: DBRCOUT DD statement output**

```
*************** TOP OF DATA **********************
INIT.DB     DBD(CUSTOMER) SHARELVL(3) TYPHALDB
/*******************************************************************************
INIT.PART   DBD(CUSTOMER) PART(CUSTP01)                                       -
  KEYSRTG(X'F1F1F1F2F2F0F9F5F1F1F1F2F2F0')                        */
  KEYSRTG(11122095111220)                                       */
  FSPF(10)                                                -
  FBFF(005)                                               -
  DSNPREFX(BMC.CUSTOMER)                                    -
CHANGE.DB   DBD(CUSTP01)  NOPINIT
/*******************************************************************************
INIT.PART   DBD(CUSTOMER) PART(CUSTP02)                                       -
  KEYSRTG(X'F1F1F1F3F3F0F9F5F1F1F1F3F3F0')                        */
  KEYSRTG(11133095111330)                                       */
  FSPF(10)                                                -
  FBFF(005)                                               -
  DSNPREFX(BMC.CUSTOMER)                                    -
CHANGE.DB   DBD(CUSTP02)  NOPINIT
/*******************************************************************************
INIT.PART   DBD(CUSTOMER) PART(CUSTP03)                                       -
  KEYSRTG(X'FFFFFFFFFFFFFFFFFFFFFFFFFFFFF')                     */
  FSPF(10)                                                -
  FBFF(005)                                               -
  DSNPREFX(BMC.CUSTOMER)                                    -
CHANGE.DB   DBD(CUSTP03)  NOPINIT
*******************************************************************************/
*************** BOTTOM OF DATA **********************
```

**Database Partition Estimate Report**

The Database Partition Estimate Report provides detailed results of a partition boundary recommendation.

The report is written to the BMCPRINT data set.

Figure 115 on page 487 shows the Database Partition Estimate report.

---

**Figure 115: Database Partition Estimate report**

```
* TOP OF DATA * DATABASE PARTITION ESTIMATE REPORT * DBD=CUSTOMER *
DATE: 02/06/2015   TIME: 15:02   DB TYPE: HIDAM
* BOTTOM OF DATA *
```
Calculating DEDB space

This section describes how to use the DEDB space calculation tool to calculate space for a DEDB area.

Overview of the DEDB space calculation tool

The DBA Toolkit DEDB space calculation tool offers a quick and convenient way to calculate disk space requirements for a Fast Path data entry database (DEDB) area before allocating space. The DEDB space calculation tool provides the following information:

- Percentage breakdown of total space used
Calculating space for a DEDB area

You can calculate space for a DEDB area in the Navigation window.

1. In the Navigation window, right-click the RECON object for which you want to calculate DEDB area space and choose **DEDB Space Calc**.

The DEDB Space Calc window (Figure 116 on page 489) is displayed.

**Figure 116: DEDB Space Calc window**

2. On the Data Entry tab, specify area parameter and SDEP (optional) values.
Building IDCAMS control statements

This section describes how to use the Build IDCAMS control statements tool to build IDCAMS DELETE, ALLOCATE, and DEFINE CLUSTER control statements for existing data sets, based on current information in the system catalog.

Overview of the Build IDCAMS control statements tool

The Build IDCAMS control statements tool offers a quick and convenient way to produce and store control statements for existing data sets.

These statements can then be used by other utilities that require data sets to be redefined during processing. The control statements created can be edited to incorporate new definition specifications when required.

This tool drives the Automatic Delete/Define (DRAMS) utility of the BMC Backup and Recovery Solution for IMS product to capture data set allocation information and create the IDCAMS DELETE, ALLOCATE, and DEFINE CLUSTER control statements.

The input control statements for this tool specify the following items:

- Selection criteria for the data sets to include
- Data sets or members in which to store generated IDCAMS control statements
- Values for a subset of DRAMS utility input parameters

The tool selects data sets from information in the Advisor registry. You can specify data sets to be selected for an IMSplex, an Advisor database group, a specific DBD, a DBD name pattern, a specific HALDB partition or DEDB area, or a HALDB partition/DEDB area name pattern. If you specify a partition name, area name, or name pattern, you must also specify a DBD or DBD name pattern.

The Build IDCAMS utility can store all generated IDCAMS statements in one output data set, or the utility can store DELETE statements in one data set and store...
ALLOCATE and DEFINE CLUSTER statements in another. The output data sets can be PDSs or sequential files. If an output data set is a PDS, the utility writes the generated IDCAMS statements for selected objects to PDS members. If the MEMBERTYPE keyword value is DBD, the member name is the DBD name (if the selected object is not a partitioned database and not a DEDB) or the name of each selected partition/area (if the selected object is a partitioned database or a DEDB). The utility places statements for all data sets that belong to a DBD or a partition/area into a single member. If the MEMBERTYPE keyword value is DDN, the utility places statements for each data set in a separate member named with its DDNAME.

The Build IDCAMS control statements tool supports the following parameters of the DRAMS utility. The tool uses internal default parameters for all other parameters.

- DELETENOSCRATCH
- GENERATEMAXCC
- OMITVOL
- USESMS

For more information about these parameters, see the section about using DRAMS utility in the Backup and Recovery Products for IMS Recovery Utility User Guide.

### Using the Build IDCAMS control statements tool

You can use the Build IDCAMS control statements tool through the console.

1. Right-click a RECON object in the navigation window, and choose **Generate JCL**.

2. In the Generate JCL window, select the row that contains the BATCH UTILITIES solution BUILD IDCAMS DELETE/DEFINE STATEMENTS feature.

3. Click **Generate JCL**.

4. In the Generate JCL popup window, select the values that you want to use for the DRAMS utility parameters.

5. Click **OK**.

The Generate JCL function creates the JCL and displays it on the JCL tab of the Generate JCL window. You can view, edit, save, and submit the generated JCL.

### JCL requirements

The following table lists batch JCL statements for the Build IDCAMS control statements tool.
Table 64: JCL statements for the Build IDCAMS control statements tool

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td>Specify PGM=CPCBATCH.</td>
</tr>
<tr>
<td>STEPLIB</td>
<td>Identify the library that contains the CPC load modules.</td>
</tr>
<tr>
<td>CPCSYSIN</td>
<td>Define the data set that contains control statements for commands and keywords that execute CPC batch functions.</td>
</tr>
<tr>
<td>CPCLOG</td>
<td>(optional) Specify the output data set to contain the CPC message log.</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>(optional) Specify the output data set to contain the DRAMS utility report.</td>
</tr>
</tbody>
</table>

Figure 117 on page 492 shows sample batch JCL for the build IDCAMS control statements tool.

```plaintext
//DELDEF1 EXEC PGM=CPCBATCH,TIME=1439
//STEPLIB      DD DISP=SHR,DSN=bmc.load.library
//DLFILE       DD DISP=SHR,DSN=YOUR.DELETE STATEMENT SAVE
//DFFILE       DD DISP=SHR,DSN=YOUR.DEFINE STATEMENT SAVE
//CPCSYSIN DD *
DELDEFN -
RECON_DSN(YOUR.RECON1) -
DBGROUP(DELDEFNGRP) -
DELFILE(DLFILE) -
DEFNFILE(DFFILE)
/*
```

**Keywords**

The following table lists keywords that are used to control processing of the Build IDCAMS control statements tool.

These keywords are specified in the CPCSYSIN control statement data set.
**Note**

A name value can be a name pattern that uses asterisks (*) as wild card characters. If you specify an asterisk as the last character in a pattern, the tool treats all remaining characters as wild card characters. If trailing spaces are required as part of the pattern, use ampersands (&) instead of asterisks.

If you specify only one of the following keywords, the tool places all IDCAMS statements in the corresponding data set:

- DELFILE
- DEFNFILE
- DELDATASET
- DEFNDATASET

If you do not specify any of these keywords, the tool places all IDCAMS statements in the CPCLOG data set.

---

**Table 65: Keywords for the Build IDCAMS control statements tool**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECON_DSN</td>
<td>RECON data set name</td>
<td>None</td>
<td>Specify the RECON data set name for the batch IDCAMS control statement tool.</td>
</tr>
<tr>
<td>DBGROUP</td>
<td>Database group name defined in Advisor</td>
<td>None</td>
<td><em>(conditional)</em> Specify a database group that contains the DBDs and partitions to be selected to build IDCAMS statements for the associated database data sets.</td>
</tr>
<tr>
<td>DBDNAME or DBD</td>
<td>Database name or DBD name pattern**</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name of the DBD or name pattern of the DBDs to be selected to build IDCAMS statements for the associated database data sets.</td>
</tr>
<tr>
<td>PARTNAME or PART</td>
<td>Partition name or partition name pattern**</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name of the partition or the name pattern of the partitions to be selected to build IDCAMS statements for the associated database data sets.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------</td>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AREANAME or AREA</td>
<td>DEDB area name or area name pattern**</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name of the area or the name pattern of the areas to be selected to build IDCAMS statements for the associated database data sets.</td>
</tr>
<tr>
<td>DELFILE</td>
<td>Output data set DDNAME contained in JCL</td>
<td>None</td>
<td><em>(conditional)</em> Specify the ddname for the DELETE statement output data set. If you specify the DELFILE keyword, do not specify the DELDATASET keyword. The DELETE statement output is placed in this data set. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records.</td>
</tr>
<tr>
<td>DEFNFILE</td>
<td>Output data set DDNAME contained in JCL</td>
<td>None</td>
<td><em>(conditional)</em> Specify the ddname for the ALLOCATE and DEFINE CLUSTER statement output data set. If you specify the DEFNFILE keyword, do not specify the DEFNDATASET keyword. The ALLOCATE and DEFINE CLUSTER statement output is placed in this data set. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DELDATASET</td>
<td>Output data set name</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name for the DELETE statement output data set. If you specify the DELDATASET keyword, do not specify the DELFILE keyword. The value specifies the name of an existing data set in which to place the DELETE statement output. The data set is dynamically allocated during processing. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records.</td>
</tr>
<tr>
<td>DEFNDATASET</td>
<td>Output data set name</td>
<td>None</td>
<td><em>(conditional)</em> Specify the name for the ALLOCATE and DEFINE CLUSTER statement output data set. If you specify the DEFNDATASET keyword, do not specify the DEFNFILE keyword. The value specifies the name of an existing data set in which to place the ALLOCATE and DEFINE CLUSTER output. The data set is dynamically allocated during processing. The data set can be sequential or partitioned. It must be defined with 80-byte fixed-length records.</td>
</tr>
<tr>
<td>DELETENOSCRATCH</td>
<td>NO, YES, BOTH</td>
<td>NO</td>
<td>See the section about DRAMS utility keywords in the <em>Backup and Recovery Products for IMS Recovery Utility User Guide</em>.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>GENERATEMAXCC</td>
<td>NO, YES, ENHANCED</td>
<td>YES</td>
<td>See the section about DRAMS utility keywords in the Backup and Recovery Products for IMS Recovery Utility User Guide.</td>
</tr>
<tr>
<td>MEMBERTYPE (or the abbreviation MBRTYPE)</td>
<td>DBD, DDN</td>
<td>DBD</td>
<td>Controls how the Build IDCAMS Control Statements tool constructs data set members to contain the created IDCAMS statements. For descriptions of the accepted values, see Accepted values for the MEMBERTYPE keyword on page 496.</td>
</tr>
<tr>
<td>OMITVOL</td>
<td>NO, YES</td>
<td>NO</td>
<td>See the section about DRAMS utility keywords in the Backup and Recovery Products for IMS Recovery Utility User Guide.</td>
</tr>
<tr>
<td>USESMS</td>
<td>NO, YES</td>
<td>NO</td>
<td>See the section about DRAMS utility keywords in the Backup and Recovery Products for IMS Recovery Utility User Guide.</td>
</tr>
</tbody>
</table>

**Accepted values for the MEMBERTYPE keyword**

You can specify one of the following values for MEMBERTYPE:

**DBD**

Build a single member for the selected object. The member contains IDCAMS statements for all data sets that belong to that object. The member name depends on the object:

- If the selected object is *not* a partitioned database or DEDB, the member name is the DBD name.
- If the selected object *is* a partitioned database or DEDB, the member name is the partition or area name.
Annotating objects

You can use the Notes window to enter free-form information about the selected object (group, database, partition, or DEDB area) in the console.

For example, you can enter the object’s nickname, unusual characteristics and uses, and reminders about actions that are related to the object.

To annotate objects

1 In the Navigation window, right-click the object for which you want to enter notes, and choose Notes.

You can enter a maximum of 256 characters. The system tracks the date and time when the note was created or last updated, and the TSO user ID of the person who added or updated the note.

An object can have only one note. If an object has a note and you enter another note, the original note is replaced. Deleting all text from a note deletes the note and all associated date, time, and user information.

A note is specific to the object to which it is attached and is not inherited from a higher level in the hierarchy.
Annotating objects
Using the JCL script language

This section describes the JCL script language that you can use with BMC database administration products for IMS. BMC provides a JCL scripting language that you can use to write custom solution scripts for your environment.

Scripts are instructions that are executed when a user generates and submits JCL with the Generate JCL function. Using the JCL script language, you can create scripts that meet the specific requirements of your environment.

The JCL scripts that BMC distributes with the database administration products for IMS are stored as follows:

- The predefined script load modules are stored in the SCRIPTLB data set. The default target data set name is hlq.DBUSCRL (if the non-merged option is selected during installation) or hlq.IMSCRL (if the merged option is selected during installation).

- The script source code is stored in the SCRIPTSC data set. The default target data set name is hlq.DBUSCRS (if the non-merged option is selected during installation) or hlq.IMSCRS (if the merged option is selected during installation).

Both data sets are specified in the CPCxINI initialization file (which is located in the sample library). The ddname that defines the SCRIPTLB data set is UTIDSN. The ddname that defines the SCRIPTSC data set is USERDSN.

During installation, both data sets are allocated and populated. You can add scripts to the SCRIPTSC data set. No restrictions apply to the number or size of JCL scripts.

At execution, source scripts are interpreted dynamically. Interpreted script listings containing syntax messages and instruction numbers are written to the SYSOUT file. If a script contains invalid syntax, processing terminates with errors; review the output file for error messages.

The following SYSOUT-type files are produced in the ADV server address space if the corresponding DD statements have been added to the ADV procedure (PROC):

- The DUELIST data set is written when a script source is used and contains interpreted script listings, including syntax messages and instruction numbers.
The DUEPRINT data set is written when a script is executed.

- The DUETRACE data set is written when you turn on tracing in the INI file.

Scripts are located for execution in the following order:

1. Load library of scripts (SCRIPTLB)
2. Source library (SCRIPTSC)

## Diagnostic tracing facility

To diagnose problems within your scripts, use the diagnostic tracing facility.

A trace is activated by insertion of the following DD statement into the JCL step that executes the script facility:

```plaintext
//DUETRACE DD SYSOUT=*  
```

Diagnostic information is written to the DUETRACE output file. Figure 118 on page 500 shows the format of diagnostic information.

### Figure 118: Diagnostic JCL format

```
----+----1----+----2----+----3----+----4----+----5----+----6----+
SCRIPT1  000001 MSG         INFO  Start of SCRIPT1  
SCRIPT1  000002 CALL        SCRIPT2  
SCRIPT2  000001 LET         VARIABLE1=1  
SCRIPT2  000002 LET         VARIABLE2=2  
SCRIPT1  000003 JCL          
  *JCL  000001 JOB00001 JOB (ACCOUNT), 'AUTHOR', MSGCLASS=X, CLASS=A  
  *JCL  000002 /*  
  *JCL  000003 /*STEP01 EXEC PGM=IEFBR14  
SCRIPT1  000004 IF          1=2  
SCRIPT1  000016 MSG         INFO  End of SCRIPT1  
```

Table 66 on page 500 lists column value definitions.

### Table 66: Column value definitions

<table>
<thead>
<tr>
<th>Column</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Name of the script being executed</td>
</tr>
<tr>
<td>11</td>
<td>Script instruction number</td>
</tr>
<tr>
<td>18</td>
<td>Script command being executed</td>
</tr>
<tr>
<td>30</td>
<td>Resolved parameters or operands for a command</td>
</tr>
</tbody>
</table>

500  Database Products for IMS Advisors and Toolkit User Guide
### Language syntax

This section discusses rules that determine JCL script language syntax.

### Free format

JCL scripts are free format.

Statements may start in any column, may have their parameters placed onto separate lines, and may share lines with other statements. Blank lines are also permitted. Figure 119 on page 501 shows the free format.

**Figure 119: Free-format JCL scripts**

```
  1  2  3  4  5  6
LET                  /* Statements start in any column
A = 1                /* Parameters may be split across lines
LET B = 2  LET C = 3 /* Multiple statements can be on 1 line
let a = 1             /* Upper or lower case are equivalent
/* Blank lines are permitted
/* Comments are preceded by ‘/*’ and apply to the rest of the line
```

JCL script statements must have their parameters specified according to the syntax and sequence that are described for that statement.

### Mixed-case text

Action statements may be written in mixed-case (uppercase and lowercase) text.

---

<table>
<thead>
<tr>
<th>Column</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>JCL (when displaying constructed JCL statements)</td>
</tr>
<tr>
<td>12</td>
<td>JCL statement sequence number</td>
</tr>
<tr>
<td>18</td>
<td>JCL statement image (after variable substitution is complete)</td>
</tr>
</tbody>
</table>

**Note**

Script instruction numbers are assigned at execution during script interpretation. The DUELIST output file contains the script list showing instruction numbers and script syntax messages.
Evaluation of script commands and other keyword parameters is the same for uppercase and lowercase text. However, text data that appears within quotes is evaluated exactly as entered.

```
LET A = 'A ' is the same as: let a = 'A'
LET A = 'A ' is the not the same as: LET A = 'a'
```

JCL data lines within a JCL GEN... ENDJCL blocks are case sensitive and are treated like text data within quotes. These lines should be written in uppercase text to avoid JCL errors. Where a variable appears within the body of the JCL data lines, the variable name may be written in mixed-case text.

**Comments**

Comments within the JCL scripts are preceded by the asterisk-slash (/*) characters.

Comments may begin at any line column and apply to the remaining portion of the line. No termination marker exists for a comment. Comments may be interspersed with script statements and their parameters.

The comment marker (/*) is ignored in the following situations:

- When embedded within textual data that is surrounded by quotes (for example, 'This /* is not a comment')
- When embedded within JCL that is contained in a JCL GEN... ENDJCL block

**Paragraph labels**

Paragraph labels are identified by a 1-character to 32-character name, ended by a colon (:).

The label name must start with an alphabetic character and may be specified at any line column. The script line containing the label can also contain other script statements.

Labels are relevant within one JCL script only; the same label name may be used in multiple scripts. The GOTO statement may reference labels to alter the processing sequence within a script.
String expressions

Most JCL script language statements use string expressions for their parameters (or operands). A string expression can include any of the following items:

- Literal value (for example, LET A = 'ABC')
- Variable (for example, LET A = &B)
- Concatenation of literal value and variable (for example, LET A = 'See ' || &SYSUID || ' for more details')

Concatenation bars

Variables and literals can be concatenated by the use of concatenation bars (||).

Two bar characters indicate that concatenation is in effect. A bar character has the hex value 4F.

Variable processing

A variable is a logical name or reference that may be assigned a value.

Variables are entries within a variable pool or dictionary service that are provided by the JCL script language facility. Most JCL script statements reference variables, and context determines whether the specified variable name or the contents (or value) are assigned the required variable.

An ampersand (&) character is used to prefix a variable name when the content or value of the variable is to be used. The ampersand is omitted when the context of the script statement requires use of a variable name.

LET VARIABLE_NAME = &VALUE_OF_THIS_VARIABLE

Simple variables

Variable names have a maximum length of 32 characters and may consist of any National character (A to Z, 0 to 9, @, #, $) and underscore (_).

Any other character terminates the variable name but is included as part of the string evaluation. Trailing blanks also terminate the variable name.
An ampersand (&) is prefixed to a variable name when referencing the value of the variable. A period (.) is a variable delimiter that is used within recursive variables. The maximum data contents for one variable is 255 bytes.

All variables (simple, recursive, and array) are globally accessible to other scripts. When a variable is written into the dictionary or variable pool, the variable remains accessible to the current JCL script and any other called JCL scripts. Variables are not deleted, although their values may be changed. A variable may be set to null (no value).

### Recursive variables

A recursive variable has a name that is constructed from a combination of other variables and string literals.

A recursive evaluation is performed to determine the variable name.

**Example**

&(&LEVEL.FLAG) is a recursive variable. Components within parentheses are evaluated first. Nested sets of parentheses are permitted and resolved by the script language interpreter to determine the variable name. Variables may be interspersed with literals to construct the variable name. For example, if &LEVEL contains the value T, &(&LEVEL.FLAG) becomes &TFLAG.

Syntax evaluation of recursive variables follows these guidelines:

- A left and right parentheses pair indicates a new level of recursive evaluation.
- Each left parenthesis must be preceded by an ampersand (&).
- The number of right parentheses must match the number of left parentheses.
- Nested evaluation conditions are permitted.
- A right parenthesis acts as a delimiter for a variable and signals the end of a variable or a string group that is to be evaluated.
- Quotation marks are not permitted within any part of a variable name.
- Concatenation bars (||) are not permitted within any part of a variable name.
- A period acts as a variable name delimiter. When the period is removed from an evaluation, variables or strings following the period are appended to the variable value.
- A resolved variable name must conform to the conventions for script variables.
- If any referenced variable is undefined, a null value is assigned to that variable.

### Array variables

An array is a table of data containing multiple values.

For example, a simple, two-dimensional array has rows of data as its first dimension. The second dimension contains columns of data. An array variable describes a specific intersection of the dimensions to access the data contents.

An array variable is constructed of the following components:

- Array name
- Dimension intersection entries

To reference the TABLE1 array, third row, and second column within the row, the array variable would be written as the following statement:

&TABLE1:3:2

The array name may be a maximum of 28 characters for a one-dimensional array, 24 characters for a two-dimensional array, and continuing to reduce by four characters for each successive dimensional level. Name size reduces as a result of the way array elements are stored within the variable pool dictionary.

Syntax evaluation of array variables follows these guidelines:

- The array name is specified first.
- Each specific dimension entry is defined in sequence, prefixed by a colon (:).
- All dimension entries must be supplied.
- A maximum of seven dimensions (or levels) is permitted for an array.
- Array name or dimension entry values may be defined as a literal, a simple variable, a recursive variable, or an array variable.
- If an array variable resolves outside the boundaries of an array, the script terminates with errors.
Array variable dimension counts

LET and DO statements provide a DIMENSION function to count the number of entries within an array.

A specific dimension may be specified, where dimension 1 counts the number of rows of data and dimension 2 counts the number of columns of data. A one-dimensional array has rows of data with only one column of values. A two-dimensional array has rows of data with multiple columns of data.

For row counts (dimension 1), the count starts at the first row and continues until no row data for the first column can be found. For column counts (dimension 2), the count starts at the first column for the first row and continues incrementing the column until no column data for the first row can be found.

DIMENSION count calculation depends upon the definition of column 1 data for all valid rows, and columnar data for all columns in row, 1 for dimension-1 counts and dimension-2 counts. If the requested array is undefined or does not contain array data, zero is returned.

Array variable deletion

An array may have all cells (or elements) deleted by redefining the array as a simple variable or an array of a different dimensional size.

Figure 120 on page 506 shows a sequence of LET statements that may be used to create array entries or to delete the array.

Figure 120: Array variable deletion statements

```
000001 LET NAMES:1 = 'BILL'      /* This defined NAMES
000002 LET NAMES:2 = 'FRED'      /* as a 1-dimensional
000003 LET NAMES:3 = 'JANE'      /* array with 3 rows.
000004
000005
000006                           /* This destroys all
000007                           /* the array elements.
000008 LET NAMES = ''            /* and creates a simple
000009                           /* variable with a null
000010                           /* value.
000011
000012
000013                           /* This destroys the
000014 LET NAMES:1 = 'BILL'      /* simple variable and
000015 LET NAMES:2 = 'GAIL'      /* creates a 1-dimensional
000016                           /* array with 2 rows.
000017
000018
000019                           /* This destroys the
000020 LET NAMES:1:1 = 'BILL'    /* 1-dimensional array and
000021 LET NAMES:1:2 = 'JONES'   /* creates a 2-dimensional
000022                           /* array with 1 row.
```
Variable processing

Variable substitution within JCL

JCL within a JCL GEN... ENDJCL block may contain a mix of literal JCL text and variables. This section discusses rules that govern variable use and substitution in JCL statements.

JCL statements that are to be prepared for submission are defined with the JCL GEN... ENDJCL block. All coding lines between JCL GEN and ENDJCL statements are considered to be input JCL.

The following rules govern processing:

- Each statement between the JCL GEN block and the ENDJCL block represents one JCL statement.
- The statement may be JCL, comments, or in-stream data statements.
- JCL syntax checking is not performed.
- JCL is processed as it appears in the statement with column positions honored. Normal JCL statements must start in column 1.
- JCL is case sensitive. Uppercase text translation is not performed. Variable names may be entered in mixed case (uppercase and lowercase) text and are resolved correctly. However, fixed data text remains in the case in which it was entered. Text values that are inserted by means of a substitution variable are not translated into uppercase text. Variable contents must be in the correct uppercase or lowercase format.
- Variable substitution is performed according to the following rules:
  — Only simple variables may be used within JCL statements. Recursive variables and array variables are not supported, as they complicate JCL coding because of data shuffling and variable delimiter issues. This limitation is overcome by adjusting the JCL script to assign recursive or array variable values to a simple variable immediately preceding the JCL script statement.
  — An ampersand (&) must prefix all variable names.
  — The presence of two ampersands (&&) causes variable substitution to be ignored. You can specify temporary data set names into your JCL.
  — The presence of an ampersand-space (as in CAT & DOG) causes variable substitution to be ignored, as no variable name is supplied.
  — Where JCL requires a textual value preceded by a single ampersand (&) without substitution (as in check &LASTCC), this requirement can be achieved...
by assigning the textual value to a variable with the LET script statement and then inserting the variable into the JCL text:

```
LET TEXT = '&LASTCC'
JCL GEN
```

To determine whether the SEARCH works, check &TEXT:

```
ENDJCL
```

— The use of special characters, a blank, or a period delimits the variable name.

— When a special character delimiter is used--for example (or)--the variable value is inserted and the special character is included as part of the text.

— When a blank character is used to delimit the variable name, the variable value is inserted followed by a blank character. Data to the right of the delimiting blank is shuffled left or right to preserve one blank character.

— When multiple blanks delimit the variable, any data to the right of the variable remains in its columnar position. Shuffling data left does not occur.

  If insufficient room exists to insert the variable value and honor trailing data to the right of the variable, the trailing data is shuffled right with one interleaving blank preserved.

— When a period (.) delimits the variable, the variable value is inserted and the period is removed from the text. To retain a period in the text, you need to specify the variable name, followed by a delimiting period, followed by a period for the normal text.

— If the inserted data causes an overflow beyond column 72, data truncation occurs. No error conditions or messages are issued.

— The variable values are substituted at the time the JCL GEN... ENDJCL statements are processed. The values used equate the variable value setting at the time of the JCL statement.

— The value of the substitution variable is inserted without case translation. If the variable is assigned lowercase text values, the resultant JCL contains lowercase text.

- Constructed JCL statements are added to a concatenated list of JCL. Multiple sets of JCL GEN... ENDJCL may be executed to build a sequence of JCL.

### System variables

During execution of a JCL script, the script can access system variables to obtain information about the job, the user, the date, and the time.
Except for LASTCC, all system variables are prefixed with SYS. Table 67 on page 509 lists system variables.

**Table 67: System variables**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;LASTCC</td>
<td>Return code set that was set by the execution of certain script statements</td>
</tr>
<tr>
<td>&amp;SYSCOMMAND</td>
<td>Name of the JCL script that was invoked initially</td>
</tr>
<tr>
<td>&amp;SYSDATE</td>
<td>Current date in the format CCYY-MM-DD</td>
</tr>
<tr>
<td>&amp;SYSDLNM</td>
<td>Literal/*</td>
</tr>
<tr>
<td>&amp;SYSDATE</td>
<td>Current Julian date in the format YYDDD</td>
</tr>
<tr>
<td>&amp;SYSJOB</td>
<td>Current job name (of the CPC/ADV region)</td>
</tr>
<tr>
<td>&amp;SYSJOBID</td>
<td>Current job number (of the CPC/ADV region), in format JOBnnnnn (or TSUnnnnn if run under TSO)</td>
</tr>
<tr>
<td>&amp;SYSJSDAT</td>
<td>Job start date (of the CPC/ADV region), in the format YYDDD</td>
</tr>
<tr>
<td>&amp;SYSSNDATE</td>
<td>Current date in the format DDMMYY</td>
</tr>
<tr>
<td>&amp;SYSNSDAT</td>
<td>Job start date (of the CPC/ADV region) in the format DDMMYY</td>
</tr>
<tr>
<td>&amp;SYSPROC</td>
<td>Current procedure (PROC) step name (of the CPC/ADV region)</td>
</tr>
<tr>
<td>&amp;SYSRC</td>
<td>Highest return code set by a MSG statement so far</td>
</tr>
<tr>
<td>&amp;SYSS#</td>
<td>Two-digit statement number assigned to the next statement in the sequence of statements that have been processed since the last job statement was processed</td>
</tr>
<tr>
<td>&amp;SYSSCRIPT</td>
<td>Name of the JCL script that is being executed</td>
</tr>
<tr>
<td>&amp;SYSSTEP</td>
<td>Current job step name (of the CPC/ADV region)</td>
</tr>
<tr>
<td>&amp;SYSSTIME</td>
<td>Current job start time (of the CPC/ADV region) in the format HHMMSS</td>
</tr>
<tr>
<td>&amp;SYSTEMID</td>
<td>Operating system identifier</td>
</tr>
<tr>
<td>&amp;SYSTIME</td>
<td>Current time in the format HHMMSS</td>
</tr>
<tr>
<td>&amp;SYSTRACE</td>
<td>Current setting for the TRACE statement (ENABLED or DISABLED)</td>
</tr>
<tr>
<td>&amp;SYSUID</td>
<td>Current job user ID or logon ID (of the CPC/ADV region)</td>
</tr>
<tr>
<td>&amp;SYSUSER</td>
<td>Current job user ID or IMS Advisor logon ID</td>
</tr>
</tbody>
</table>

**CALL statement**

The CALL statement can be used to invoke other scripts.
This usage allows scripts to be developed in a modular way. SCRIPT and EXIT parameters determine the processing sequence if the called script cannot be located.

**Syntax**

The CALL statement uses the following syntax:

```
CALL SCRIPT = string
CALL EXIT = string
```

**Examples**

Following are examples of the CALL statement:

```
CALL SCRIPT = JOBCARD
CALL SCRIPT = &JCL01
CALL EXIT = JOBCARD
CALL EXIT = JOBCARD
```

**Description**

CALL SCRIPT and CALL EXIT keywords are fixed and must be specified as shown.

The string can be a literal or variable that becomes the name of the script to be invoked. The resolved script name must conform to standards for PDS member names.

Using the CALL statement, you can develop your scripts in a modular way. You can place commonly used scripting statements into a separate script member so that multiple caller scripts are able to share them.

The CALL SCRIPT statement must be satisfied. The resolved script name must be located; otherwise the script terminates with errors. The CALL EXIT command does not have to be satisfied. If the script that is named in the CALL EXIT command is not located, processing continues at the next instruction in the script.

Called scripts are located by first searching the library of compiled script load modules. If a script is not located here, the library of source scripts is searched. If the script is not located, processing terminates or continues, based on the use of the SCRIPT or EXIT command parameters.
CALL PROGRAM statement

The CALL PROGRAM statement allows transfer of control to a program.

Syntax

The CALL PROGRAM statement uses the following syntax:

```
CALL PROGRAM = string
PARM= ( string )
```

Examples

Following are examples of the CALL PROGRAM statement:

```
CALL PROGRAM = BMCPGM1
CALL PROGRAM = BMCPGM1 PARM=(&PARM1 'NEXTPARM')
IF &LASTCC > 4
  Error handling script statements
ENDIF
```

Description

The invoked program must be located from a job’s normal STEPLIB data set list.

The program is invoked by using the standard assembler calling convention. Parameters in the PARM list are passed as follows:

- Each parameter string is evaluated, and the result is stored left-justified in a 255-byte blank-padded string that is preceded by a one-byte size indicator.

- The called program can reference or modify parameter strings.

- On return, if a parameter string was a single defined variable (by means of LET, SUBSTR, or TRANSLATE statements), the modified string (stripped of trailing spaces) becomes the new value of the variable.

Upon program completion, the script continues running at the instruction immediately following the CALL statement. The program’s return code is available.
in the system variable LASTCC. If a program abend occurs, LASTCC is set to 9999 and the abend code is available from the system variable LASTABEND.

DO statement

A DO loop executes script statements iteratively within a DO ... ENDDO block, incrementing a control variable until a predetermined value is reached.

Syntax

The DO statement uses the following syntax:

```
DO variable =start value — TO end value
| Dimension (array, number) |
| BY increment |
```

Examples

Following are examples of the DO statement:

```
DO COUNTER = 1 TO 10
| Dependent Script Statements |
ENDDO

LET START = 1
LET END = 10
DO COUNTER = &START TO &END
| Dependent Script Statements |
ENDDO

DO COUNTER = 1 TO 5 BY 2
| Dependent Script Statements |
ENDDO
```
Description

The DO statement sets up a block of script statements that are processed iteratively.

The block of statements is bounded by the DO and ENDDO statement pair. DO loops can be nested.

Loop processing is controlled by a control variable. When the DO statement is processed, the control variable is assigned the start value, which can be a literal or a variable. The end value and incremental values are also evaluated. Start, end, and incremental values must be zero or a positive integer; otherwise, an error condition occurs and script processing terminates. If the DO statement is not specified, the incremental value defaults to 1.

The DO statement end value can also be specified on the DIMENSION parameter. The DIMENSION parameter determines the number of entries within a dimensional level of an array, and the DO loop is performed until this number is reached. If the requested array is undefined or does not contain array data, zero is returned. Because the DIMENSION parameter uses the actual name of the array, the parameter is not prefixed by an ampersand (&).

If a dimensional level number is not specified, the first dimension is assumed. If the first dimension is selected, the DO loop end value is based on the number of rows within the array. If the second dimension is selected, the end value is based on the number of columns that are associated with the first row of the array.

The DO loop dependent statements are not executed if the start value exceeds the resolved end value. DO X1 = 1 TO 1 causes the statements within the DO ... ENDDO block to be executed once. DO X1 = 1 TO 0 causes the statements within the DO ... ENDDO block to be bypassed.

If the DO loop is executed, at the end of the DO loop the control variable is compared to the end value. If the control variable is greater than or equal to the end value, DO loop processing stops; otherwise, the control variable increments and processing resumes at the beginning of the DO loop.

The control variable value can be altered within the loop to hasten completion of DO loop processing. Changes to end or incremental variables do not affect DO loop controls, as values that were determined at the start of DO loop processing are used.

Looped statements cannot contain a GOTO statement that references a label outside the DO loop. If this situation occurs, the script terminates with errors.
GOTO statement

The GOTO statement branches unconditionally to the label that is named by the statement.

Syntax

The GOTO statement uses the following syntax:

```
GOTO label
```

Examples

Following are examples of the GOTO statement:

```
GOTO THE_END
THE_END:
   LET LABEL='THE_END'
   GOTO &LABEL
   THE_END:
```

Description

The GOTO statement transfers control to the nominated label.

In the GOTO statement, the label can be specified as a literal or a variable. The label must be a literal followed by a colon (:).

The GOTO statement cannot branch to a label outside a DO ... ENDDO block. If the statement tries to branch to a label outside a DO ... ENDDO block, the script terminates with an error message.

IF statements

IF statements cause transfer of control, based on the evaluation of a condition.
Syntax

IF statements use the following syntax:

```
IF string operator string ( string )
```

Dependent script statements

```
ELSE
```

Dependent script statements

```
ENDIF
```

Examples

Following are examples of IF statements:

```
IF &NAME=&LAST_NAME
   LET COUNT=&LAST_COUNT
ELSE
   IF &NAME IN ('FRED' 'JANE')
      IF 5< &COUNT
         LET COUNT=1
      ENDIF
   ENDIF
ENDIF
```

Description

IF ... ENDIF statements define a block of script statements that are executed conditionally. IF sets the condition with subsequent dependent scripts statements that are executed if the condition is true. An ELSE statement defines the block of script statements for execution if the condition is false.

An IF statement compares one string to another. These strings can be literals or variables. An operator dictates the type of comparison to be performed. The following operators are valid:

- EQ
- =
Any of these operators can be preceded by NOT. The NOT symbol (\^ for PC keyboards; ¬ for IBM terminal keyboards) has the hex value X'5F'.

The IN operator allows a list of comparative string values to be entered. The operators must be specified within parentheses. These values are compared to determine whether a true condition or a false condition exists.

If the result of IF comparison is true, processing continues with the statements dependent upon the IF ... ENDFJ block. The ELSE statement is optional. Use the ELSE statement to specify processing statements if the condition is false. An ENDFJ statement must be used to identify the end of the IF processing block.

**JCL GEN statements**

JCL GEN ...ENDJCL statements identify a block of JCL statements to be prepared for submission.

**Syntax**

JCL GEN statements use the following syntax:

```
JCL GEN

JCL cards, code starting in column 1

ENDJCL
```
Example

Following are examples of JCL GEN statements:

```
JCL GEN
//&JOBN    JOB (&ACCOUNT)'SCRIPT JCL',CLASS=&JOBCLASS,
// NOTIFY=&SYSUID
//*
//*This is some sample JCL with embedded variables for
//*substitution. The JCL is coded starting in column1.
//*just as if it was in a JCL deck. Note some of the
//*rules that govern how the JCL is treated:
//*
//* &TEMPFILE      No variable substitution because of the &&
//* CAT & DOG      No variable substitution—the & is followed by a space.
//* DSN(&MBR)      Special characters like the bracket delimit the variable.
//* &SYSJOB..FILE A (.) delimits the variable and is removed from the text.
ENDJCL
```

Description

JCL GEN statements to be prepared for submission are defined with the JCL GEN ... ENDJCL block. All coding lines between JCL GEN and ENDJCL statements are considered to be input JCL.

Following are some key considerations:

- **Case translation**
  Translation to uppercase text is not performed. Text data is treated exactly as it is entered. This rule also applies to variable data that is inserted into the JCL. Enter JCL the way you want it to be displayed.

- **Variable types**
  Only simple variables can be used. Recursive or array variables must be changed to simple variables before they are used in the JCL.

- **Variable delimiters**
  Non-alphanumeric characters like parentheses or commas delimit a variable name, and the special character is retained in the resolved JCL. A period (.) is a delimiter, but it is removed from the resolved JCL.

- **Columns and shuffling**
  When multiple blanks delimit a variable, any data to the right of the variable remains in its columnar position. If not enough room exists to insert the variable value or if only one blank space was used as a delimiter, one interleaving space is retained and the remaining data is shifted left.
## LET statement

The LET statement assigns a value to a variable.

### Syntax

The LET statement uses the following syntax:

```plaintext
LET string = string

string arithmetic operator string

DIMENSION (array, number)

DATATYPE (string)

LENGTH (string)
```

### Examples

Following are examples of the LET statement:

```plaintext
LET VARIABLE   = 'ABC' /*Example 1
LET MESSAGE    = 'MY NAME IS'|| &SYSUID /*Example 2
LET COUNT      = &COUNT+1 /*Example 3
LET DIM        = DIMENSION (NAME_ARRAY,1) /*Example 4
LET Dtype      = DATATYPE (&VARIABLE) /*Example 5
LET len        = LENGTH (&VARIABLE) /*Example 6
LET &DBD.FLAG     ='Y' /*Example 7
LET &DBD|| 'FLAG'  ='Y'
LET VARIABLE   = 'COUNTER' /*Example 8
LET &VARIABLE  = 1
```

Example 1 shows a simple variable assignment. Example 2 shows the use of concatenation bars to set the assigned value 'MY NAME IS' followed by the system variable SYSUID. Example 3 shows an arithmetic operation. Example 4 shows a method of returning the count of rows or columns within an array. Example 5 shows a method of determining whether a variable contains numeric or character data. Example 6 shows a method of determining the length of a variable or literal string. Example 7 shows two methods of constructing the receiving variable name by combining a variable value with a literal. Example 8 shows a method of specifying the receiving variable name by means of the contents of another variable.
Description

The LET statement assigns a value to a variable.

The string that provides the value can be a literal, a variable, or concatenations of both. This string is evaluated, and the result is placed into the receiving variable. The receiving variable can be named by the contents of a variable (for example, LET &VAR_NAME = '). The receiving variable name is stored in the variable VAR_NAME.'

Arithmetic calculations require an arithmetic operator and an additional literal or variable. Multiple arithmetic conditions can be used within one LET statement. The following operands are supported:

- Add (+)
- Subtract (-)
- Divide (/)
- Multiply (*)
- Integer division (%)
- Remainder (~)

Use the DIMENSION parameter to determine the number of entries within a specific dimensional level of an array. The receiving variable is assigned a numeric value corresponding to the number of entries that are found. If the requested array is undefined or does not contain array data, zero is returned. Because the DIMENSION parameter uses the actual name of an array, it is not prefixed by an ampersand (&).

If a dimensional level number is not specified, the first dimension is assumed. If the first dimension is selected, the receiving variable contains the number of rows within the array. If the second dimension is selected, the returned count represents the number of columns that are associated with the first row of the array.

Use the DATATYPE parameter to determine the data type of a string expression. The receiving variable resolves to 'NUM' if the string expression is numeric (contains only numbers, with an optional operand prefix). If the string expression is not numeric, the receiving variable resolves to 'CHAR'.

MSG statements

MSG statements issue a message to the JCL script report log and set the script return code.

If an error message is issued, script processing is terminated immediately.
**Syntax**

MSG statements use the following syntax:

```
MSG INFO string
MSG WARN string
MSG ERROR string
```

**Examples**

Following are examples of MSG statements:

- `MSG INFO 'Messages are written to DDNAME BMCPRINT'`
- `MSG INFO 'Processing for '|| &DBD|| ' is now complete.'`
- `MSG WARN 'Return code set to 04, but processing continues'
- `MSG ERROR 'Processing stops with return code 08'`

**Description**

MSG statements allow JCL scripts to write information, warning, or error messages to an output file with the ddname BMCPRINT.

These messages serve information or diagnostic purposes.

The MSG INFO statement is useful for reporting status information. The MSG INFO statement does not affect the script completion return code or processing sequence.

The MSG WARN statement also writes a message, but maximum return code for the JCL script is increased to 04. The JCL script continues processing.

The MSG ERROR statement writes an error message, sets the JCL script return code to 08, and immediately stops script processing.

*Table 68 on page 521* lists JCL script completion codes.
Table 68: JCL script completion codes

<table>
<thead>
<tr>
<th>Completion code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Processing completed successfully.</td>
</tr>
<tr>
<td>04</td>
<td>Processing completed successfully; however, a MSG WARN message was issued.</td>
</tr>
<tr>
<td>08</td>
<td>Processing terminated prematurely because of a MSG ERROR statement.</td>
</tr>
<tr>
<td>12</td>
<td>Processing terminated abnormally because of a script error that was detected by the script language interpreter; an error message indicates the cause of the failure.</td>
</tr>
</tbody>
</table>

The message text string can be literals, variables, or concatenations of both. The maximum message text length is 255 bytes. If the resolved value of the message text exceeds 255 bytes, truncation occurs.

**SCRIPT statements**

SCRIPT statements identify a JCL script.

**Syntax**

SCRIPT statements use the following syntax:

```
SCRIPT scriptname
```

```
Dependent JCL script statements
```

```
ENDSCRIPT
```

**Examples**

Following are examples of SCRIPT statements:

```
SCRIPT TESTIT
  MSG INFO 'This script has the name TESTIT'
ENDSCRIPT
```
Description

The SCRIPT statement identifies the beginning of a JCL script and assigns a script name.

The script name must conform to the naming convention for PDS data set members. The ENDSNRIPT statement designates the end of the script.

Because SCRIPT statements must precede all other script statements, only comment lines or blank lines can precede SCRIPT statements. All lines following the ENDSNRIPT statement are ignored.

SEARCH statement

The SEARCH statement scans array data to find a row with a specific set of column values.

Syntax

The SEARCH statement uses the following syntax:

```
SEARCH ARRAY array name
FIND (column number column value)
FROM start index
```

Examples

Following are examples of the SEARCH statement:

```
LET INDEX = 0
SEARCH ARRAY NAME_TABLE
  FIND (1 &SYSUID 'FRED' 'JANE'
   2 +10 +20)
  FROM INDEX
```
Description

The SEARCH statement scans a one or two-dimensional array for a row with a particular set of column values.

The row value of a single dimension array is treated as if it were the first column within the row.

Array rows to be scanned are limited by the start index as named in the FROM clause. The start index specifies the starting point for the search and must contain a numeric value. The search begins at the next row entry and continues until a match is found or until the array is exhausted. To begin the search at the start of the array, set the start index variable to zero.

SEARCH compares the column number and column value parameters that are entered in the SEARCH statement with the data at the current array row, as determined by the start index. To compare multiple columns, specify the column numbers in numeric sequence.

When the SEARCH statement specifies multiple values for a column number, that column is considered to be a match if any value equates with the array row value and the column value. Columns that are omitted from the SEARCH statement are ignored for matching purposes. If the column value for comparison is numeric, it must be enclosed in quotes.

Each array row is evaluated in turn. The search is successful when each column number and column value combination in the SEARCH statement matches the values in the array row. The search is complete when a match is found or the last remaining array row is processed.

After execution of the SEARCH statement, the system variable LASTCC contains a completion code and the start index value may be updated to reflect the array row that satisfied the search.

Table 69 on page 524 lists SEARCH completion codes.
Table 69: SEARCH completion codes

<table>
<thead>
<tr>
<th>Completion code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;LASTCC=0</td>
<td>The search was successful. A matching array row was found. The start index variable that is specified in the FROM clause is updated to reflect the matching row number.</td>
</tr>
<tr>
<td>&amp;LASTCC=4</td>
<td>The search was unsuccessful. A matching array row was not found. The start index variable that is specified in the FROM clause remains unchanged.</td>
</tr>
</tbody>
</table>

**SORT statement**

The SORT statement sorts a one-dimensional array or a two-dimensional array.

**Syntax**

The SORT statement uses the following syntax:

```
SORT ARRAY array name
  BY (column number)
  INTO array name 2
  REMOVE DUPLICATES
```

**Examples**

Following are examples of the SORT statement:

```
SORT ARRAY NAME_TABLE BY (1,2)
SORT ARRAY NAME_TABLE BY (1 C A 2 C D)
SORT ARRAY NAME_TABLE BY (1) INTO WORK_TABLE
SORT ARRAY NAME_TABLE BY (1) INTO WORK_TABLE REMOVEDUPLICATES
```
**Description**

The SORT statement sorts a one-dimensional array or a two-dimensional array.

The row value of a one-dimension array is treated as if it were the first column within the row.

Sort sequence is determined by the BY parameter. Each BY parameter specifies a combination of column number, data type (C or N), and ascending/descending sequence option (A or D). You can enter multiple column number combinations, where the first number is considered the primary sort key and other numbers are minor sort keys.

The column data type is indicated by the character C for alphanumeric data or N for numeric data. Numeric data can include a sign character. For sort purposes, leading zeros are suppressed. Null numeric column data is interpreted as a zero value. If character data type is not specified, it is assumed.

Ascending or descending sort sequence is indicated by the character A or D. If sort sequence is not specified, ascending sort sequence is assumed.

The INTO keyword indicates whether to place the resultant sorted array data into a new array, identified by array name 2. If the INTO keyword is not specified, the original array is replaced with the sorted array.

REMOVEDUPLICATES signifies that the resultant array should contain only one row for each unique combination of sorted columns. As the sort columns are evaluated, rows with duplicate values in all sort columns are discarded, retaining only the first detected row for those values. When discarding array rows, values within non-sorted columns are not considered. The accepted row position within the original array is unpredictable and cannot be assumed to be the first occurrence within the original array.

**SUBSTR statement**

The SUBSTR statement assigns part of a string expression to a variable.

**Syntax**

The SUBSTR statement uses the following syntax:
Examples

Following are examples of the SUBSTR statement:

```
SUBSTR NAME_PREFIX = &NAME (1,4)
SUBSTR NAME_PREFIX = &NAME (1,4,'')
SUBSTR MSG_TEXT_NAME = 'User name:'      (1,15,'')
SUBSTR MSG_TEXT_PHONE = 'Phone Number:'  (1,15,'')
```

Description

The SUBSTR statement examines a string from a given start position and copies its contents, character by character, into a receiving variable.

The examination string can be a literal, a variable, or concatenations of both. The number of characters that are copied is specified in the length parameter. Trailing blanks are stripped unless a pad character is specified.

If the number of characters to be copied exceeds the number of characters that are available from the originating string, the pad character is copied until the length parameter is satisfied. If no pad character is specified, SUBSTR operation stops after the last non-blank character is copied.

The pad character must be a literal. Pad characters can include special characters or a space (blank) by enclosing the character within quotes.

TRACE statement

The TRACE statement translates variable data to a new value that is based on a list of arguments.

Syntax

The TRACE statement uses the following syntax:
Example

Following are examples of the TRACE statement:

```
TRACE DISABLE
CALL SCRIPT = PASSWORD           /*Verify product or services passwords
CALL SCRIPT = SECURITY           /*Security check user access
TRACE ENABLE
```

Description

Use the TRACE statement to limit output that is written to the script diagnostic tracing facility.

Diagnostic tracing becomes active when you add the trace output DD statement to script execution JCL. The TRACE DISABLE statement deactivates this tracing. Statements occurring after the TRACE DISABLE statements are not displayed in the diagnostic trace output until the TRACE ENABLE statement is executed, which allows sensitive script code to be hidden from users. The TRACE statement has no other effect on the execution of script statements.

If the TRACE statement is placed within a DO loop or IF statement, the TRACE statement is executed conditionally. The TRACE option (disabled or enabled) remains in force until its setting is changed by another TRACE statement. Subsequent TRACE statements do not have to exist within the same script. A TRACE DISABLE statement that is activated in one script can be reversed by a TRACE ENABLE statement in another subroutine script.

The system variable SYSTRACE returns the value of ENABLED or DISABLED to indicate the current TRACE statement setting. Even if the TRACE setting is ENABLED, tracing does not occur unless the trace output DD statement has been added to the script execution JCL.
TRANSLATE statement

The TRANSLATE statement translates variable data to a new value that is based on a list of arguments.

Syntax

The TRANSLATE statement uses the following syntax:

```
TRANSLATE name = compare string (compare value replacement value)
```

Examples

Following are examples of the TRANSLATE statement:

```
TRANSLATE ROLE = &NAME ('FRED' 'Project Manager')
('JOHN' 'Team Leader')
(&NAME 'Programmer')
```

Description

The TRANSLATE statement compares the compare string value to entries within a translation list.

Each translation list entry contains a compare value and a replacement value pair. Each translation list compare value is assessed in turn. When a match is found, the replacement value is placed into the receiving variable (which is identified by name) and translation stops. If no match is found, the receiving variable is assigned a null value.

To ensure that a match is found in the translation list, you can create an entry in which the compare value is the same as the compare string.

Compare string, compare value, and replacement value parameters can be literals, variables, or concatenations of both.
JCL script examples

This section presents examples of JCL scripts.

Figure 121 on page 529 shows script example 1. Figure 122 on page 530 shows script example 2. Figure 123 on page 531 shows script example 3. Figure 124 on page 531 shows script example 4. Figure 125 on page 532 shows script example 5.

Figure 121: JCL script example 1

```jcl
000001 /******************************************************************
000002 */
000003 /* Script: EXAMPLE   A sample script to illustrate the JCL generator.
000004 /* It is a primary script, not a sub-routine script.
000005 */
000006 /* Author      YYYYMMDD Description
000007 /* ----------------------------------------------
000008 /* JCL Writer 20000401 Initial creation as a sample JCL script
000009 /******************************************************************
000010 script example
000011
000012 /*-----------------------------------------------------------------
000013 /*~ 1. Perform initialization processing
000014 /*-----------------------------------------------------------------
000015 call script = @first    /* sets up variables common to all scripts
000016 call script = @jobcard  /* builds and inserts the job statement
000017 call script = @steplib  /* builds an array of STEPLIB datasets
000018
000019 /*-----------------------------------------------------------------
000020 /*~ 2. Build the body of the JCL
000021 /*-----------------------------------------------------------------
000022 jcl gen                             /* 1st step deletes files
000023
000024 //DELETE EXEC PGM=IEFBR14
000025 /*REPTFL1 DD DSN=&SYSUID..BMCS$.SCRIPT.&SYSJOBID..WORK1.
000026 // DISP=(MOD,DELETE),UNIT=&F_DASD_UNIT,
000027 // DCB=(LRECL=133,BLKSIZ3=13300,RECFM=FBA),
000028 // SPACE=(TRK,(1,1))
000029 endjcl
000030
000031 substr dd = &sysndate (1,2)        /* DD in DDMMYY
000032 translate monthly_report_reqd = &dd /* Monthly report required?
000033      ('01' 'YES') /* YES if 1st day of month
000034      (&dd 'NO')  /* NO for all other days
000035 jcl gen                             /* 2nd step is user program
000036
000037 //REPORT EXEC PGM=PROGRAM1,PARM=('DEFINE','REPORT')
000038 /*REPTFL1 DD DSN=&SYSUID..BMCS$.SCRIPT.&SYSJOBID..WORK1.
000039 // DISP=(NEW,CATLG),UNIT=&F_DASD_UNIT,
000040 // DCB=(LRECL=133,BLKSIZ3=13300,RECFM=FBA),
000041 // SPACE=(TRK,(3,1))
000042 endjcl
000043
000044 jcl gen /* 2nd step is user program
000045
000046 endjcl
000047 let stepdd = 'STEPLIB'               /* set the DDname to STEPLIB
000048 do x1 = 1 to dimension(steplib)    /* loop thru each entry.
000049 jcl gen                             /* Build JCL for STEPLIBs
000050 //&STEPDD DD DISP=SHR,DSN=STEPLIB:&X1
000051 endjcl
000052 let stepdd = ' '                   /* Reset the DDname to blank
000053 enddo
```

Appendix A Using the JCL script language 529
Figure 122: JCL script example 2

```plaintext
script @first

 /*-----------------------------------------------------------------
 /*~ 1. Set up common working variables
 /*-----------------------------------------------------------------
 let f_dasd_unit    = 'SYSDA'            /* The DASD unit device
 let f_tape_unit    = 'CART'             /* The tape unit device
 let f_cart_unit    = 'CART'          /* The cartridge unit device
 let f_sysout_class = '*'                /* SYSOUT=? on DD statement
 let f_job_account= 'ACCOUNT CODE'     /* Job card ACCOUNT value
 let f_job_class    = 'A'                /* Job card CLASS value
 let f_job_msgclass = 'X'                /* Job card MSGCLASS value
 let f_job_notify   = &sysuid            /* Job card NOTIFY value
 let f_user_lib     = 'CUSTOMER.LOADLIB' /* DSN for user programs
 let f_db2_lib      = ' '                /* DSN for DB2 (not used)
 let f_ims_lib      = 'IMS610.LOADLIB'   /* DSN for IMS
 let f_system_lib   = 'SYS1.LINKLIB'     /* DSN for system programs

 /*-----------------------------------------------------------------
 /*~ 2. Log a message indicating that processing has started
 /*-----------------------------------------------------------------
 substr hh = &systime (1,2)     /* the HH in HHMMSS
 substr mm = &systime (3,2)     /* the MM in HHMMSS
 substr ss = &systime (5,2)     /* the SS in HHMMSS
 let time = &hh || ':' || &mm || ':' || &ss
 msg info 'JCL Script '  
    &syscommand
    ' invoked by user ''
    &sysuid
    '"' on ' 
    &sysdate
    ' at ' 
    &time
 endscript
```
Figure 123: JCL script example 3

```assembly
000001 /**********************************************************************
000002 /*
000003 /* Script: @JOBCARD  Build the JCL job card.
000004 /*
000005 /* * Author YYYYMMDD Description
000006 /*
000007 /* JCL Writer 20000401 Initial creation to build the JCL job card
000008 /**********************************************************************
000009 script @jobcard
000010
000011 /*~ 1. Add JOB, ACCOUNT and NAME parameters
000012 /*~
000013 /*--------------------------------------------
000014 /*
000015 substr jobn = &sysuid || 'JCLSCRIPT' (1,8)
000016
000017 jcl gen
000018 //&JOBN    JOB (&F_JOB_ACCOUNT),'BMC JCL SCRIPT',
000019 //         NOTIFY=&SYSUID,
000020 //         CLASS=&F_JOB_CLASS,MSGCLASS=&F_JOB_MSGCLASS
000021 endjcl
000022
000023 endscript
```

Figure 124: JCL script example 4

```assembly
000001 /**********************************************************************
000002 /*
000003 /* Script: @STEPLIB  Create an array of STEPLIB datasets.
000004 /*
000005 /* * Author YYYYMMDD Description
000006 /*
000007 /* JCL Writer 20000401 Initial creation to build an array of STEPLIBs
000008 /**********************************************************************
000009 script @steplib
000010
000011 /*~ 1. Set up STEPLIB a 1 dimensional array
000012 /*~
000013 /*--------------------------------------------
000014 /*
000015    let x1 = 1                           /* Set index pointer to 1
000016
000017 if &f_db2_lib = ' '                   /* If DB2 is used
000018     let steplib:&x1 = &f_db2_lib        /* set the DSN name
000019     let x1 = &x1 + 1                    /* increment array pointer
000020 endif
000021
000022 if &f_ims_lib = ' '                   /* If IMS is used
000023     let steplib:&x1 = &f_ims_lib        /* set the DSN name
000024     let x1 = &x1 + 1                    /* increment array pointer
000025 endif
000026
000027 if &f_system_lib = ' '                /* If a system lib is used
000028     let steplib:&x1 = &f_system_lib     /* set the DSN name
000029     let x1 = &x1 + 1                    /* increment array pointer
000030 endif
000031
000032 if &f_user_lib = ' '                  /* If a user lib is used
000033     let steplib:&x1 = &f_user_lib       /* set the DSN name
000034     let x1 = &x1 + 1                    /* increment array pointer
000035 endif
000036 ```
Figure 125: JCL script example 5

```plaintext
000001 /******************************************************************
000002 /* Script: @LAST      Write out script completion message. 
000003 /* Author     YYYYMMDD Description
000004 /* ---------- -------- --------------------------------------------
000005 /* JCL Writer 20000401 Initial creation to write out a completion msg
000006 /******************************************************************
000007 script @last
000008 /*-----------------------------------------------------------------
000009 /*~ 1.  Log a message indicating that processing has completed
000010 /*-----------------------------------------------------------------
000011 substr hh = &systime (1,2)     /* the HH in HHMMSS
000012 substr mm = &systime (3,2)     /* the MM in HHMMSS
000013 substr ss = &systime (5,2)     /* the SS in HHMMSS
000014 let time = &hh || ':' || &mm || ':' || &ss
000015 msg info 'JCL Script ' 
000016       &syscommand 
000017       ' completed RC=' 
000018       &sysrc 
000019       on ' 
000020       &sysdate 
000021       ' at ' 
000022       &time
000023 endscript
```
Index

A

ACBLIBs
Auto Configure 121
Activate Analysis parameter 260
Activate Conditional Reorg parameter 257
Activate Fast Path Space Data Collection parameter 258
Activate FF Hourly Collect parameter 258
active object information in View Activity window 420
administrator privileges 96
ADS not available condition 242
ADS not defined condition 242
ADV repositories 73
ADV server 73, 75
ADVISE command 406
Alloc Failed parameter 252
Alloc Freespace % parameter 200, 230, 247
Alloc Freespace(LS) % parameter 230
Allow Reorg parameter 257
Allow Reorg, conditional reorganization 398
ALTRCN DD statement 177
AMSOUT Data Set Name parameter 266
analysis, data collection 141
Analyze CA Group to Trigger ICs parameter 256
annotating objects 497
Area is not set to preopen parameter 242
Area Stopped parameter 252
Area Used % parameter 230
AREA USED PERCENT EXCEEDED exception 152
AREANAME keyword 352
Assets entry is not cataloged parameter 242
AUDIT function 170
AUDIT schedule 281
authentication, z/OS 104
Auto Configure

ACBLIBs 121
DBDLIBs 122
IMS subsystems 120
populated information 118
RECONs 120
automated data collection 140, 141
automatic refresh in Monitor Recovery windows 414
automatic refresh rate 414

B

Backout is needed parameter 242
BACKOUT NEEDED exception 178, 195
Backup and Recovery Solution (for IMS) 411
Backup Tape Unit Name parameter 273
base values, viewing historical 165
batch reports 444
BCSS 73
starting 75
BDS group 75
object in multiple 242
benefits 65
BMC Consolidated Subsystem 73
BMC Primary Subsystem 73
BMC product synergy 67
BMC utilities 195
BMC-provided solutions 326
BMCP 73
starting 75
BRIADVSE member of sample library 177
BRIUBSOL program 341
browser, installing console 97
Build Unload During Reorg parameter 266
bypass image copy feature 390
BYPASS keyword 401
CA DATA SET NOT CATALOGED exception
CA Extract model parameter
CA File Size Threshold parameter
CA group, object not in any run
CA RUN IN ERROR exception
CA Splits % parameter
CA Splits Count parameter
CAP CA Model parameter
CAP Global Options parameter
Capture Statistics parameter
challenges of data management
change accumulation data set model
change accumulation parameters
Change Accumulation utility
change accumulation, no CA run in range
Checkpoint data set model parameter
CI Splits % parameter
CI Splits Count parameter
CI Splits parameter
CLASS keyword
client, console
Collecting data
Compare History reports
Completion Code parameter
Completion Code Type parameter
Compression Technique parameter
Conditional Image Copy feature
Conditional Image Copy Simulation utility
Conditional Reorg batch report
configuration, mainframe components
connection
  defining
  Enterprise List
  personal list
console
  exiting
  implementing TLS
installation
installing
launching
uninstalling
user privileges

Consolidate E-mail messages parameter
control block disassembly tool, batch
conventions, documentation
COUNT keyword
CPC
  starting
CPCBATCH program
CPCBATCH utility
control statement syntax
CPCSSID keyword
CPCSYSIN control statement syntax
CPCTSCHD program
Critical parameter
data collection
  automated
  BMC utilities
  return codes
Data Collection Interval parameter
data management
  challenges
data management goals
Data Set Name Prefix parameter
Data set size (tracks) parameter
Database Advisor
  benefits
  components, mainframe
  database maintenance cycle
  Fast Path DEDB support
  logical structure
  mainframe components
  threshold parameters
  workflow
Database Advisor solution process
Database Availability parameter
Database Exception report
database maintenance cycle, traditional
Database Partition Estimate Report
database recovery monitoring
Days to Retain Tape parameter
DB command
DBD Cmd Pending parameter
DBD Deleted by OLCHNG parameter
DBD Freespace Low parameter
DBD In Error parameter
DBD keyword
DBD/Area Init Fail parameter
DBD/Area Locked parameter 251
DBD/Area Not Open parameter 251
DBD/Area Stopped parameter 252
DBDLIBs
  Auto Configure 122
DBDS group
  object not in any 242
DBDS IN MULTIPLE DBDS GROUPS exception 179
DBDS NOT IN ANY DBDS GROUP exception 180
DBDS NOT IN CHANGE ACCUM GROUP exception 180
DBGROUP keyword 352
DBR Cmd Pending parameter 252
DBRC Auth Fail parameter 252
DBRCOUT DD statement output 486
DBSYNCH schedule 281
DBUPHAL member 477
DD Names parameter 260
Dead parameter 253
DEDDB support 74
Detail Record Retention (days) parameter 273
Details table, Exception List 303, 305
disaster recovery 409
Display Messages window 108
DLIAANL2 477
DLIAAUDR program 170
DLIATRPT utility 454
  control statements 455
  example reports 457
  JCL 454
documentation information 21
DOVF CI's less than CI's per UOW % parameter 230
DOVF Space Utilization % parameter 240
DSN SIZE EXCEEDS THRESHOLD exception 203
DSNPREFIX keyword 480
Dual Output Image Copy Mask Name parameter 268
dynamic allocation models 268, 276
environment reports 444
error conditions, stand-alone conditional reorg 403
ESTIMATE job 478
EXCEPLST command 311
exception
  IMAGE COPY NEEEDED 184
Exception List
  Details table 303, 305
generating JCL 334, 335
  menu commands 308
  status commands 305
  Summary table 303, 305
exceptions
AREA USED PERCENT EXCEEDED 152
BACKOUT NEEDED 178, 195
CA DATA SET NOT CATALOGED 179
CA RUN IN ERROR 179
clearing Recovery Advisor exceptions from the repository 195
DBDS IN MULTIPLE DBDS GROUPS 179
DBDS NOT IN CHANGE ACCUM GROUP 180
DSN SIZE EXCEEDS THRESHOLD 203
EEQE(S) FOUND 180, 195
FSE ERROR 152
GENMAX BELOW THRESHOLD 181
GENMAX EXCEEDED THRESHOLD 181
GRPMAX BELOW THRESHOLD 181
GRPMAX EXCEEDED THRESHOLD 182
HALDB PARTITION NOT INITIALIZED 182
HIDAM PRIMARY INDEX ERROR 152
IC DATA SET IN ERROR 182
IC RECOMMENDED BY CHANGE ACCUM 183, 195
IMAGE COPY NEEDED 195
IMAGE COPY RECOMMENDED 184, 195
INCORRECT CI SIZE 153
INSUFFICIENT FREE SPACE 203
MAXIMUM OSAM ALLOC EXCEEDED 156
MAXIMUM VSAM ALLOC EXCEEDED 156
MIN # OF CA RUNS NOT AVAILABLE 185
MIN # OF IC'S NOT AVAILABLE 185
MORE LOGS THAN THRESHOLD 186
NO AREA DATA SET AVAILABLE 186
NO CA RUN WITHIN TIME RANGE 187, 195
NO IC WITHIN TIME RANGE 187, 195
NO SECIC WITHIN TIME RANGE 188, 195
NO SECLOG WITHIN TIME RANGE 188
NO SECSLD WITHIN TIME RANGE 189
NO VALID IMAGE COPY 189, 195
POINTER ERROR 156
PREOPEN NOT SET FOR AREA 189
PRILOG ENTRY IN ERROR 190
PRILOG ENTRY NOT CATALOGED 190
PRILOG SIZE EXCEEDS THRESHOLD 157, 204
PRISLD ENTRY IN ERROR 190
PRISLD ENTRY NOT CATALOGED 191
PROHIBIT AUTHORIZATION FOUND 191
READ ONLY 191
RECLAIMABLE FREESPACE EXCEEDED 157
RECLAIMABLE IOVF EXCEEDED 157
RECLAIMABLE SPACE EXCEEDED 158
RECON DISCARDED / UNAVAILABLE 204
RECOVERY NEEDED 192, 195
RECOVERY NEEDED COUNT INVALID 158, 192
RECOVPD BELOW THRESHOLD 192
RECOVPD EXCEEDED THRESHOLD 192
SECLOG ENTRY IN ERROR 193
SECLOG ENTRY NOT CATALOGED 193
SECSLD ENTRY IN ERROR 194
SECSLD ENTRY NOT CATALOGED 194
TOO FEW DOVF CONTROL INTERVALS 158
TOO FEW EMPTY RAPS 158
TOO FEW IOVF BLOCKS 159
TOO MANY CA SLETS 204
TOO MANY CI SLETS 205
TOO MANY EXTENTS 205
TOO MANY VOLUMES 206
UNALLOCATED DATASET 162
UNINITIALIZED DATASET 162
Exclude all objects parameter 242
Exclude nonrecoverable objects parameter 242
Expand Unload parameter 264
Expected RAA Growth % parameter 230
export exception data 309
Extents parameter 200, 247
Extract Data Set Mask Name parameter 272
EXTRACT job 477

F

Fast Path DEDB support 74
Fast Path Member Type parameter 260
FEATURE keyword 352
FF COLLECT HOURLY schedule 281
FF COLLECT schedule 281
Filter Objects dialog box 425
FP COLLECT schedule 281
FP SPACE COLLECT schedule 281
Fragmentation % parameter 230
Fragmentation parameter 230
Free Blocks % parameter 230
From parameter 253
FSE ERROR exception 152
Full Function Member Type parameter 260
function return codes 287

G

gathering data 140, 141
GDG parameter 141
generated JCL 340
Index 537

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

generating JCL, Exception List 334, 335
GENMAX BELOW THRESHOLD exception 181
GENMAX EXCEEDED THRESHOLD exception 181
GENMAX value 242
global options module
  Change Accumulation utility 276
  Image Copy utility 268
  Recovery utility 276
goals of data management 39
Growth in Record I/O % parameter 230
GRPMAX BELOW THRESHOLD exception 181
GRPMAX EXCEEDED THRESHOLD exception 182
GRPMAX value 242

H

HALDB Member Type parameter 260
HALDB modeling and migration tools 472, 476
HALDB partition is not initialized parameter 242
HALDB PARTITION NOT INITIALIZED exception 182
HDAMGROW keyword 480
Help
  online 21
HIDAM PRIMARY INDEX ERROR exception 152
Historical Analysis Wizard 61, 459
historical base values 165
HMS Recall Pending parameter 251
Honor DBD Free Space Definition % parameter 230
hosts
  connecting 111
  defining 111
HTML Format parameter 253

I

I/O Probability Growth % parameter 230
IC DATA SET IN ERROR exception 182
IC DATA SET NOT CATALOGED exception 183
IC RECOMMENDED BY CHANGE ACCUM exception 183, 195
IC Triggering by CA feature parameters 256
IC-NEEDED status 242
IC-RECOMMENDED status 242
ICP Global Options parameter 268
ICP Image Copy Model parameter 268
ICPVSMIC utility 395
IDCAMS Define Control Statements parameter 260
IDCAMS Delete Control Statements parameter 260
image copy bypass feature 390
image copy count threshold 242
image copy data set model 268, 276
Image Copy Indexes parameter 266
Image copy is needed parameter 242
Image copy is recommended parameter 242
image copy missing or invalid 242
IMAGE COPY NEEDED exception 184, 195
IMAGE COPY RECOMMENDED exception 184, 195
Image Copy utility 195
Improvement % parameter 230
IMS Not Available parameter 251
IMS Status Monitoring parameters 250
IMS subsystems
  Auto Configure 120
Include SMS Classes in JCL parameter 260, 264
Include user image copies parameter 242
INCORRECT CI SIZE exception 153
Index Term 24, 102, 447
Indirect List Data Sets (ILDS) parameter 276
Input Image Copy Mask Name parameter 272
installation
  client requirements 96
  console 97
  locations 96
  mainframe components 72
INSUFFICIENT FREE SPACE exception 203
interfaces, user
  console 107
Invoke Compression parameter 272
IOVF Space Utilization % parameter 230
IOVF Threshold Type parameter 266
IOVF Threshold Value parameter 266
IOVF UOW's less than total UOW's % parameter 230
IRMADVSE program 176
Issue WTO's parameter 253

J

JCL Generation utility 350
JCL generation, parameters that influence 332
JCL Generator. See Utility Migration JCL Generator
JCL Generator, saving JCL 340
JCL script 326, 340
Job Card 1 parameter 260
Job Card 2, 3, 4 parameter 260
Job Details area in Monitor Recovery window 416
JOBCARD keyword 352
Jobs List table in Monitor Recovery window 416

K

KEYSIZE keyword 480
keywords
AREANAME 352
BYPASS 401
CLASS 352
COUNT 480
CPCSSID 401
DBD 352, 401, 479
DBGROUP 352
DSNPREFIX 480
FEATURE 352
HDAMGROW 480
JOBCARD 352
KEYSIZE 480
LASTFREE 480
MEMBER 352, 466
MERGEDSN 480
NOPINIT 480
ODS 352, 466
OFILE 352, 466
OSAM8GB 480
OUTDATASET 352, 466
OUTFILE 352, 466
PARMBLK 268, 276
PART 401, 479
PARTLIST 479
PARTMASK 480, 484
PARTNAME 352
PARTSEL 480
PARTSET 479
RECON_DSN 352
RECON1 401
REUSPRTS 480
RNAME 352
SIZE 480
SKIPFREQ 479
SOLUTION 352
TYPE 480
USERRC 401

L

Large Region parameter 260
LASTFREE keyword 480
log count more than maximum 242
Log Data Set Name Prefix parameter 266
Log Extract model parameter 276
log information in View Activity window 420
Log to Tape parameter 266
Log UNIT parameter 266
Logging parameter 266

M

mainframe components 72
manual refresh in Monitor Recovery windows 415
Maximum Monthly Records parameter 273
MAXIMUM OSAM ALLOC EXCEEDED exception 156
Maximum Stats Detail Record parameter 273
MAXIMUM VSAM ALLOC EXCEEDED exception 156
Maximum Yearly Records parameter 273
Medium Region parameter 260
MEMBER keyword 352, 466
MERFEDSN keyword 480
Messages window 108
migration to HALDB 472, 476
MIN # OF CA RUNS NOT AVAILABLE exception 185
MIN # OF IC’S NOT AVAILABLE exception 185
Minimum Base I/O Probability % parameter 230
modeling HALDBs 472, 476
models
change accumulation data set 276
image copy data set 268
image copy data sets 276
Monitor Recovery function 409
accessing 413
active object information 420
automatic refresh rate 414
considerations 411
filtering object list 425
input log information 420
JES Viewer function 415
Job Details area 416
job step activity information 419
job step failure information 423
Jobs List table 416
list of recovery job steps 416
log information 420
manual refresh 415
object information 424
Object Phase Summary area 417
options for windows 414
pausing automatic refreshes 414
processing 410
purpose of 409
resuming automatic refreshes 414
time range for listing job steps 415
View Activity window 419
working with 411
MORE LOGS THAN THRESHOLD exception 186

N

Navigation window 107
NO AREA DATA SET AVAILABLE exception 186
NO AREA DATA SET DEFINED exception 186
No area data set is available parameter 242
No area data set is defined parameter 242
No CA run exists parameter 242
NO CA RUN WITHIN TIME RANGE exception 187, 195
NO IC WITHIN TIME RANGE exception 187, 195
NO SECIC WITHIN TIME RANGE exception 188, 195
No SECLOG entry exists parameter 242
NO SECLOG WITHIN TIME RANGE exception 188
No secondary image copy exists parameter 242
No SECSDL entry exists parameter 242
NO SECSDL WITHIN TIME RANGE exception 189
NO VALID IMAGE COPY exception 189, 195
No valid image copy exists parameter 242
non-administrator privileges 96
nonrecoverable objects, excluding from Recovery Advisor 242

NOPINIT keyword 480
not-cataloged conditions 242
Notes window 497
notification parameters 253
Notify DBRC of Image Copy parameter 268
Number of CA runs is less than required parameter 242
Number of Copies parameter 268
Number of logs since last IC or CA exceeds a value parameter 242
Number of RVP IC Copies parameter 276
Number of valid image copies is less than required parameter 242

O

object information in View Objects window 424
Object is in multiple DBDS groups parameter 242
Object is in prohibit authorization status parameter 242
Object is in read-only status parameter 242
Object is not in any CA group parameter 242
Object is not in any DBDS group parameter 242
Object Phase Summary area in Monitor Recovery window 417
objects
  annotating 497
ODS keyword 352, 466
OFILE keyword 352, 466
online Help 21
Online Regions Not Avail parameter 251
OSAM Data Set Size parameter 230
OSAM Extents parameter 230
OSAM Volumes parameter 230
OSAM8GB keyword 480
OUT_DSNAME keyword 352
OUTDATASET keyword 352, 466
OUTFILE keyword 352, 466
Output Image Copy Mask Name parameter 268

P

parameters
AA Larger by % 230
Activate Analysis 260
Activate Conditional Reorg 257
Activate Fast Path Space Data Collection 258
Activate FF Hourly Collect 258
Alloc Failed 252
Alloc Freespace % 200, 230, 247
Allow Reorg 257
AMSOUT Data Set Name 266
Analyze CA Group to Trigger ICs 256
Area is not set to preopen 242
Area Stopped 252
Area Used % 230
Assets entry is not cataloged 242
Backout is needed 242
Backup Tape Unit Name 273
BuildUnload During Reorg 266
CA Extract model 276
CA File Size Threshold 256
CA Splits 230
CA Splits % 200, 247
CA Splits Count 230
CAP CA Model 276
CAP Global Options 276
Capture Statistics 273
change accumulation 276
Checkpoint data set model 276
CI Splits 230
CI Splits % 200, 247
CI Splits Count 230
Completion Code 257
Completion Code Type 257
Compression Technique 268
Consolidate E-mail messages 253
Critical 253
Data Collection Interval 258
Data Set Name Prefix 268
Data set size (tracks) 200, 247
Database Advisor 230
Database Availability 253
Days to Retain Tape 273
DBD Cmd Pending 252
DBD Deleted by OLCHNG 251
DBD Freespace Low 253
DBD In Error 252
DBD/Area Init Fail 251
DBD/Area Locked 251
DBD/Area Not Open 251
DBD/Area Stopped 252
DBR Cmd Pending 252
DBRC Auth Fail 252
DD Names 260
Dead 253
Detail Record Retention (days) 273
DOVF CI's less than CI's per UOW % 230
DOVF Space Utilization % 240
Dual Output Image Copy Mask Name 268
e-mail 253
EEQE(s) are detected 242
Exclude all objects 242
Exclude nonrecoverable objects 242
Expand Unload 264
Expected RAA Growth % 230
Extents 200, 247
Extract Data Set Mask Name 272
Fast Path Member Type 260
Fragmentation 230
Fragmentation % 230
Free Blocks % 230
From 253
Full Function Member Type 260
GDG 268
Growth in Record I/O % 230
HALDB Member Type 260
HALDB partition is not initialized 242
HMS Recall Pending 251
Honor DBD Free Space Definition 242
HTML Format 253
I/O Probability Growth % 230
ICP Global Options 268
ICP Image Copy Model 268
IDCAMS Define Control Statements 260
IDCAMS Delete Control Statements 260
Image Copy Indexes 266
Image copy is needed 242
Image copy is recommended 242
Improvement % 230
IMS Not Available 251
IMS Status Monitoring 250
Include SMS Classes in JCL 260, 264
Include user image copies 242
Indirect List Data Sets (ILDS) 276
Input Image Copy Mask Name 272
Invoke Compression 272
IOVF Space Utilization % 230
IOVF Threshold Type 266
IOVF Threshold Value 266
IOVF UOW's less than total UOW's % 230
Issue WTO's 253
Job Card 1 260
Job Card 2, 3, 4 260
Large Region 260
Log Data Set Name Prefix 266
Log Extract model 276
Log to Tape 266
Log UNIT 266
Logging 266
Maximum Monthly Records 273
Maximum Stats Detail Record 273
Maximum Yearly Records 273
Medium Region 260
Minimum Base I/O Probability % 230
No area data set is available 242
No area data set is defined 242
No CA run exists 242
No SECLOG entry exists 242
No secondary image copy exists 242
No SECSLD entry exists 242
No valid image copy exists 242
notification 253
Notify DBRC of Image Copy 268
Number of CA runs is less than required 242
Number of Copies 268
Number of logs since last IC or CA exceeds a value 242
Number of RVP IC Copies 276
Number of valid image copies is less than required 242
Object is in multiple DBDS groups 242
Object is in prohibit authorization status 242
Object is in read-only status 242
Object is not in any CA group 242
Object is not in any DBDS group 242
Online Regions Not Avail 251
OSAM Data Set Size 230
OSAM Extents 230
OSAM Volumes 230
Output Image Copy Mask Name 268
Perform IC if "nn" days since last IC 257
Pointer Validation Type 272
PRILOG Record Size % 200, 247
Primary and Secondary Indexes 276
Processing Mode 268
RAA Free Space % 230
RAA Smaller by % 230
RAA Space Utilization % 240
RAA Total Free Space % 230
Read Errors 252
Reclaimable IOVF for UOW % 230
RECON Backup Data Set Name 247
RECON Data Set Contention 247
RECON Detail Record Retention (days) 247
RECON threshold 247
RECON Warning Days 247
recovery 276
Recovery Failures Email Notification 253
Recovery is needed 242
Recovery needed count is invalid 242
Recovery threshold 242
Reorg Options 266
Repository Catalog Data Set Name 273
Repository Group Code 273
Required CI Size 230
Resolution Option 247
Retention Period 260, 264, 268
Retry Time Period 247
Return Code 247
REXX EXEC 253
Root Segments in Overflow % 230
Run Analysis After IBM HD Load 260
RVP Global Options 276
RVP IC Model 276
SDEP Freeblks Low % 251
SDEP Pointer Validation Type 272
SDEP Space Utilization % 230
Segments Split to Different Block % 230
Select any member of CA group for IC if "nn" days since last IC 256
Select members of the CA group by DBD or Partition 256
Shadow Data Set Suffix 266
Shadow Define Control Statements 266
Shadow Delete Control Statements 266
Skip IC if no updates 257, 392
Small Region 260
SMS Data Class 260, 264, 268
SMS Management Class 260, 264, 268
SMS Storage Class 260, 264, 268
SMTP Name 253
SMTP Sysout Class 253
Snapshot Option 268
Sort Work UNIT Name 260
Space Allocation Type 260
Space vs. Performance Weight Factor 253
Split Segment % 230
STA Cmd Pending 252
Subject 253
SYSOUT Class 260
Too Many RDWRT Fails 252
Total Freespace % 230
UNIT Name 264, 268
Unload Data Set Mask Name 264
Unload Data Set Name Prefix 264
Use DBRC 260
Use RVP Automatic Restart 276
Use SMS 268
Use SMS for Work Files 260
Use Tape 264, 268
User Variables 260
Verify GENMAX value 242
Verify GRPMAX value 242
Verify RECOVPD value 242
Virtual Data Set Name 268
Virtual Image Copy 268
Volumes 200, 247
VSAM Data Set Size 230
VSAM Extents 230
VSAM Volumes 230
Warning 253
Warning Days 230
Warning Days - CI & CA Splits 230
Warning Days - Data Set size 230
Warning Days - Database Performance 230
Warning Days - Extents and Volumes 230
Warning Days - Free Space 230
When CA file size threshold is exceeded 256
Work File High-Level Qualifier 260
Work File UNIT Name 260
Write Errors 252
WTO 253
XBM Subsystem ID 268
PARMBLK keyword 268, 276
PART keyword 401, 479
PARTITION-INIT-NEEDED status 242
PARTLIST keyword 479
PARTMASK keyword 480, 484
PARTNAME keyword 352
PARTSEL keyword 480
PARTSET keyword 479
pause refresh in Monitor Recovery windows 414
Perform IC if "nn" days since last IC parameter 257
POINTER ERROR exception 156
Pointer Validation Type parameter 272
populated information
Auto Configure 118
POPUPxx keyword 356
PREOPEN NOT SET FOR AREA exception 189
PRILOG ENTRY IN ERROR exception 190
PRILOG ENTRY NOT CATALOGED exception 190
PRILOG Record Size % parameter 200, 247
PRILOG SIZE EXCEEDS THRESHOLD exception 157, 204
Primary and Secondary Indexes parameter 276
PRISLD ENTRY IN ERROR exception 190
PRISLD ENTRY NOT CATALOGED exception 191
privileges
administrator 96
non-administrator 96
Processing Mode parameter 268
PROHIBIT AUTHORIZATION FOUND exception 191
PROHIBIT-AUTHORIZATION status 242
publications, related 21
PURGE schedule 281

R

RAA Free Space % parameter 230
RAA Larger by % parameter 230
RAA Smaller by % parameter 230
RAA Space Utilization % parameter 240
RAA Total Free Space % parameter 230
Read Errors parameter 252
READ ONLY exception 191
READ-ONLY status 242
RECLAIMABLE FREESPACE EXCEEDED exception 157
RECLAIMABLE IOVF EXCEEDED exception 157
Reclaimable IOVF for UOW % parameter 230
RECLAIMABLE SPACE EXCEEDED exception 158
RECON Backup Data Set Name parameter 247
RECON Data Set Contention parameter 247
RECON Detail Record Retention (days) parameter 247
RECON DISCARDED / UNAVAILABLE exception 204
RECON Reorg utility
RECON Backup Data Set Name parameter 247
RECON Data Set Contention parameter 247
RECON State report 434
RECON threshold parameters 247
RECON Warning Days parameter 247
RECON_DSN keyword 352
RECON1 keyword 401
RECONS
Auto Configure 120
RECOV-NEEDED status 242
Recovery Advisor
excluding objects from analysis 242
Alloc Freespace % parameter 200, 247
CA splits % parameter 200, 247
CI splits % parameter 200, 247
clearing exceptions from the repository 195
Data set size (tracks) parameter 200, 247
database maintenance cycle 44
Extents parameter 200, 247
logical structure 47
Monitor Recovery function 409
parameters 242, 247
PRILOG Record Size % parameter 200, 247
process explanation 173
RECON Detail Record Retention (days) parameter 247
RECON Warning Days parameter 247
return codes 178, 287
running with a batch job 176
schedule 281
Volumes parameter 200, 247
workflow 47
RECOVERY COLLECT schedule 281
Recovery Failures Email Notification parameter 253
Recovery is needed parameter 242
recovery monitoring function 409
RECOVERY NEEDED COUNT INVALID exception 158, 192
Recovery needed count is invalid parameter 242
RECOVERY NEEDED exception 192, 195
recovery parameters 276
Recovery threshold parameters 242
Recovery utility 195, 410
Recovery Wizard 370
RECOVPD BELOW THRESHOLD exception 192
RECOVPD EXCEEDED THRESHOLD exception 193
RECOVPD value 242
refresh rate in Monitor Recovery windows 414
related publications 21
Reorg Options parameter 266
REP BACKUP schedule 281
reports
  batch 444
  Compare History 441
  Database Exception 311
  RECON State 434
  Threshold Settings 454
repositories
  introduction 73
Repository Catalog Data Set Name parameter 273
Repository Group Code parameter 273
Required Group Code parameter 230
Resolution Option parameter 247
resume refresh in Monitor Recovery windows 414
Retention Period parameter 260, 264, 268
Retry Time Period parameter 247
Return Code parameter 247
return codes
  data collection 287
  Recovery Advisor 178, 287
REUSPRTS keyword 480
REXX EXEC parameters 253
REXX EXEC SCHEDULE 281
RNAME keyword 352
Root Segments in Overflow % parameter 230
RSRC command 346
Run Analysis After IBM HD Load parameter 260
RVP Global Options parameter 276
RVP IC Model parameter 276

S

saving JCL 340
schedules
  AUDIT 281
  DBSYNCH 281
  E-MAIL CONSOLIDATION 281
  ENDOFDAY 281
  FF COLLECT 281
  FF COLLECT HOURLY 281
  FP COLLECT 281
  FP SPACE COLLECT 281
  PURGE 281
  RECOVERY Collect 281
  REP BACKUP 281
  REXX EXEC SCHEDULE 281
SDEP Freeblks Low % parameter 251
SDEP Pointer Validation Type parameter 272
SDEP Space Utilization % parameter 230
SECLOG entry conditions 242
SECLOG ENTRY IN ERROR exception 193
SECLOG ENTRY NOT CATALOGED exception 193
secondary image copy missing or invalid 242
SECLOG entry conditions 242
SECSLD ENTRY IN ERROR exception 194
SECSLD ENTRY NOT CATALOGED exception 194
Segments Split to Different Block % parameter 230
Select any member of CA group for IC if "nn" days since last IC parameter 256
Select members of the CA group by DBD or Partition parameter 256
servers, starting 75
Shadow Data Set Suffix parameter 266
Shadow Define Control Statements parameter 266
Shadow Delete Control Statements parameter 266
Show jobs field in Monitor Recovery windows 415
SIZE keyword 480
Skip IC if no updates parameter 257, 392
SKIPFREQ keyword 479
Small Region parameter 260
smart image copy feature 390
SMARTIC keyword 387, 394
SMS Data Class parameter 260, 264, 268
SMS Management Class parameter 260, 264, 268
SMS Storage Class parameter 260, 264, 268
SMTP Name parameter 253
SMTP Sysout Class parameter 253
Snapshot Option parameter 268
solution definitions 326
SOLUTION keyword 352
solution process
available solutions 324
concepts 323
Solution utility 341
solutions
maintaining 379
adding user-defined solutions 381
deleting user-defined solutions 383
Solution utility 341
user-defined 330
viewing 379
Sort job 478
Sort Work UNIT Name parameter 260
Space Allocation Type parameter 260
Space vs. Performance Weight Factor parameter 253
Split Segment % parameter 230
STA Cmd Pending parameter 252
status types, Exception List 305
Subject parameter 253
Summary table, Exception List 303, 305
SUMRYRPT command 445
synergy with BMC products 67
SYSOUT Class parameter 260
T
Threshold Settings report 454
control statements 455
default 455
element reports 457
JCL 454
TOO FEW DOVF CONTROL INTERVALS exception 158
TOO FEW EMPTY RAPS exception 158
TOO FEW IOVF BLOCKS exception 159
TOO MANY CA SPLITS exception 204
TOO MANY CI SPLITS exception 205
TOO MANY EXTENTS exception 205
Too Many RDWRT Fails parameter 252
TOO MANY VOLUMES exception 206
Total Freespace % parameter 230
traditional database maintenance cycle 41
TYPE keyword 480

U
UIM Server 73
overview 96
starting 75
UNALLOCATED DATASET exception 162
UNINITIALIZED DATASET exception 162
uninstallation, console 104
UNIT Name parameter 264, 268
Unload Data Set Mask Name parameter 264
Unload Data Set Name Prefix parameter 264
update, console 103
USE command 344
Use DBRC parameter 260
Use RVP Automatic Restart parameter 276
Use SMS for Work Files parameter 260
Use SMS parameter 268
Use Tape parameter 264, 268
user image copies, including for Recovery Advisor 242
User Interface Middleware (UIM) server 96
user options, console 109
user privileges
console 96
User Variables parameter 260
user-defined solutions 330
adding 381
deleting 383
USERRC keyword 401
Utility Migration JCL Generator
   generate JCL 369
   overview 359
   starting 361

V
   Verify GENMAX value parameter 242
   Verify GRPMAX value parameter 242
   Verify RECOVPD value parameter 242
   View Activity window 419
   View Failures window 423
   View Objects window 424
   Virtual Data Set Name parameter 268
   Virtual Image Copy parameter 268
   Volumes parameter 200, 247
   VSAM Data Set Size parameter 230
   VSAM Extents parameter 230
   VSAM Volumes parameter 230

W
   Warning Days - CI & CA Splits parameter 230
   Warning Days - Data Set Size parameter 230
   Warning Days - Database Performance parameter 230
   Warning Days - Extents and Volumes parameter 230

X
   XBM Subsystem ID parameter 268

Z
   z/OS authentication 104