MainView Administration Guide

Supporting

Version 6.1.01 of MainView Infrastructure
MainView products

September 2016
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

**Address**
BMC SOFTWARE INC
2101 CITYWEST BLVD
HOUSTON TX 77042-2827 USA

**Telephone**
1 713 918 8800
or
1 800 841 2031

**Fax**
1 713 918 8000

Outside United States and Canada

**Telephone**
+01 713 918 8800

**Fax**
+01 713 918 8000

© Copyright 2004-2016 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.

BookManager, BookMaster, CICS, DB2, IBM, IBM z, IMS, MQSeries, MVS, OS/390, RACF, RMF, S/390, VTAM, WebSphere, z/OS, z/VM, zEnterprise, and zSeries are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Linux is the registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

UNIX is the registered trademark of The Open Group in the US and other countries.

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 UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES.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, 2101 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 11
Related publications ........................................................................................................... 11
Conventions ......................................................................................................................... 12

### Chapter 1 Managing MainView 13
Understanding the MainView architecture .................................................................... 13
- BBI .............................................................................................................................. 13
- Coordinating address spaces ................................................................................. 15
- Product address spaces ........................................................................................... 15
- User address spaces ................................................................................................. 17
- Runtime Component System (RTCS) ................................................................... 17
Administrator tasks ............................................................................................................ 17

### Chapter 2 Managing the coordinating address space (CAS) 21
Starting a CAS ..................................................................................................................... 21
- CAS START command parameters ....................................................................... 22
- Examples of CAS START commands ................................................................... 26
- CAS initialization in a multisystem environment ............................................... 26
Managing the CASPERM registry .................................................................................... 29
- Creating a CASPERM registry data set ............................................................... 29
- Enabling a CASPERM registry .............................................................................. 31
- Disabling a CASPERM registry ............................................................................. 31
- Changing a registry data set name ........................................................................ 32
Stopping a CAS ................................................................................................................... 32
Restarting a CAS ................................................................................................................. 34
- After applying maintenance ................................................................................... 34
- When you change the CAS libraries ................................................................. 34

### Chapter 3 Managing the z/OS product address space (PAS) 37
Understanding z/OS PAS operations ............................................................................. 37
Starting the z/OS PAS ....................................................................................................... 39
- z/OS PAS started task parameters ......................................................................... 39
- Examples of z/OS PAS START commands ......................................................... 47
About CPM control statement members .................................................................. 48
About IPM control statement members .................................................................. 50
Running z/OS PASs concurrently ................................................................................ 52
Adding a logger definition member ................................................................. 131
Editing a logger definition member ............................................................... 131
Deleting a logger definition member ............................................................ 132
Managing entries in a logger definition member ........................................... 132
Managing the log files .................................................................................. 139
Displaying a list of log files ........................................................................... 139
Displaying logged messages .......................................................................... 140
Locating messages by criteria ...................................................................... 143
Locating messages by time .......................................................................... 144
Issuing operator commands ........................................................................ 145
Exporting log records ................................................................................... 146
Controlling MainView Logger operation ....................................................... 146

Chapter 8 Managing targets and target contexts 149
Overview of targets ...................................................................................... 149
Managing targets ......................................................................................... 150
Managing target contexts ............................................................................ 151
Target context definition .............................................................................. 151
Managing target context definitions ............................................................. 159
Displaying the status of targets and systems ............................................... 176
Understanding types of Plex Manager status views .................................... 177
Accessing Plex Manager status views .......................................................... 178
Navigating with Plex Manager status views ................................................ 180
Securing contexts ........................................................................................ 181

Chapter 9 Managing SSI contexts 183
Types of SSI contexts ................................................................................... 185
Planning considerations ................................................................................ 185
SSI context definition views .......................................................................... 186
Creating an SSI context definition ............................................................... 187
Managing SSI context definitions ................................................................. 190
Managing SSI context definition members (CONDEFL view) .................... 191
Displaying an SSI context definition member ............................................... 192
Deleting an SSI context definition member .................................................. 192
Viewing products and targets in a definition member (CONDEF view) ....... 193
Adding a new SSI context definition member .............................................. 193
Viewing SSI context definition details (CONDEFD view) ............................. 194
Editing an SSI context definition ................................................................. 195
Copying an SSI context definition member ................................................. 195
Adding an SSI context definition ................................................................. 196
Changing an existing SSI context definition ............................................... 197
Canceling modifications to an SSI context definition ........................................ 198
Deleting an SSI context definition ..................................................................... 199
Installing changes to SSI context definitions ..................................................... 200
Saving an SSI context definition ....................................................................... 200
Deploying SSI context definitions ..................................................................... 201
Displaying the status of SSI contexts .................................................................. 202
Viewing context and product availability (CONACTZ view) .............................. 203
Viewing target status in an SSI context (CONACT view) ................................... 205
Viewing an individual target status (CONACTD view) ..................................... 206
Securing contexts ............................................................................................... 208

Chapter 10  Managing historical data sets ............................................................ 209
Understanding the historical database ............................................................... 209
What is the historical database? .......................................................................... 210
Historical data in a multisystem, multiproduct environment .............................. 210
When is historical data recorded? ....................................................................... 211
Where is historical data recorded? ...................................................................... 212
How many historical data sets should you allocate? ....................................... 213
Allocating historical data sets to the historical database .................................. 213
Defining historical data support .......................................................................... 214
Managing the historical database ....................................................................... 215
Viewing the status of historical data sets (DSLIST view) ................................. 216
Understanding the status of a historical data set .............................................. 217
Changing recording eligibility .......................................................................... 218
Selecting the next data set for recording .......................................................... 218
Restricting read and write access to the active data set .................................... 220
Reinstating read and write access to a quiesced data set .................................. 222
Closing a data set ............................................................................................... 223
Emptying a historical data set ........................................................................... 224
Adding a historical data set dynamically ............................................................ 225
Saving dynamic changes to the historical database ......................................... 227
Deallocating a historical data set ....................................................................... 227
Archiving and retrieving historical data ............................................................. 228

Chapter 11  Managing the MainView Infrastructure common registry ............... 231
Mounting a file system ....................................................................................... 231
Identifying a file system to a CAS ....................................................................... 232
Activating a file system ...................................................................................... 233
Other file system tasks ....................................................................................... 233
Accessing the HFSPATH view ............................................................................ 234
Chapter 12  Managing the User Interface Middleware (UIM) server  235
Creating a UIM server ............................................................... 235
  Creating a startup configuration member ............................ 236
  Creating a started task procedure ..................................... 237
  Defining the MainView Infrastructure CAS connection (optional) 239
  Setting up the HFS data set .................................................. 240
  Configuring the UIM server .................................................. 243
  Operating the UIM server .................................................... 252
  Troubleshooting UIM server problems .............................. 258

Chapter 13  Controlling diagnostic messages  261
Displaying the DIAGMSG view .............................................. 261
Understanding message and diagnostic options .................. 262
Understanding the scope of an option ................................. 266

Chapter 14  Job status information  269
Displaying job status information ........................................... 269

Chapter 15  Creating online Help  273
Writing your own Help text .................................................... 273
How MainView locates your Help text ................................. 275
Introduction to the MainView tag language ......................... 275
  Using the MainView tag language .................................... 276
  MainView tag language rules ............................................ 277
  Using control words ....................................................... 277
Commonly used tags ............................................................. 278
  Using lists ........................................................................ 279
Creating free-form Help topics .............................................. 282
  Setting up hypertext links between topics ....................... 282
MainView tags, attributes, and control words .................... 283

Appendix A  BBI diagnostic facilities  291
Messages .............................................................................. 292
LOGREC ............................................................................... 292
Dumps .................................................................................. 293
  Types produced .............................................................. 293
  BBI dump parameters .................................................... 294
Summary of BBI abend diagnostics ................................. 295

Appendix B  Journal and image log record formats  297

Contents  9
Journal log record formats ................................................................. 297
Image log record formats ................................................................. 298

Appendix C  BMC Subsystem Services  301
Overview of BBXS ................................................................. 301
Using BBXS with multiple BMC products  ........................................ 302
Installing BBXS ................................................................. 302
Initializing BBXS ................................................................. 303

Appendix D  Working with DB2 Component Services (DBC)  305
Working with the DBC subsystem ........................................ 305
  Started task for the DBC subsystem ........................................ 305
  DBC startup parameters .................................................. 307
Stopping the DBC subsystem ............................................. 316
DBC commands ................................................................. 317
DBC security ................................................................. 320
  Controlling access to DBC resources .................................. 321
  Configuring RACF security for the DBC subsystem ............... 322
DBC security parameters .................................................. 325

Appendix E  OMVS segment requirements and ESM definitions  331
General OMVS segment requirements ...................................... 331
OMVS segment requirements for TCP/IP communication .......... 338
Defining an OMVS segment to security ..................................... 339
  RACF ................................................................. 340
  CA ACF2 .............................................................. 340
  CA Top Secret (TSS) .................................................. 340

Appendix F  Suppressing U4095 abends  343

Glossary  345
About this book

This book contains detailed information about the associated product or products. This preface explains the special conventions that the book uses, and how to access related publications.

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)


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: `testsyst/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.
Managing MainView

This chapter discusses concepts that you need to understand in order to manage your MainView products, including MainView Infrastructure and its components.

Understanding the MainView architecture

MainView is an integrated family of performance management and automation products that monitor and control traditional and parallel mainframes.

MainView comprises performance monitors, automated operations, and automation applications.

MainView product integration provides host system monitoring and automation (even in remote locations) through a common terminal session, using the MainView Selection Menu. The integration of MainView products is provided through intercommunications technology known as BBI.

BBI

BBI integrates the MainView performance products within a common communications framework that operates across multiple machines in multiple locations, as shown in the figure below. This integrated architecture allows a single terminal session, using one or more MainView products, to monitor and manage multiple local or remote targets. These targets can include IBM z/OS, z/VM, or z/Linux (sysplex and nonsysplex) or subsystems like IBM CICS, IBM DB2, IBM IMS, IP, Linus Torvalds Linux, The Open Group UNIX, UNIX System Services, IBM VTAM, IBM WebSphere, and WebSphere MQ.
The BBI architecture provides a built-in separation of the data, application, and end-use dimensions of systems management for maximum flexibility and extensibility. BBI communications, data collection, and the end-user terminal session run in three distinct address spaces:

- Coordinating address space (CAS)
- Product address space (PAS)
- User address space (UAS)

This multiple-address-space structure provides a consistent, flexible environment for managing literally hundreds of systems and subsystems. Depending on the products that are installed, you can:
Access different systems and products by using target switching, direct hyperlinks between products, or multiple concurrent views on one terminal session

Summarize data on a single system or across multiple system images

Display historical or real-time data from multiple systems summarized into one view

Enter commands for multiple products on multiple systems

Apply simple or complex data filtering conditions

Coordinating address spaces

The coordinating address space (CAS) runs as a subsystem and is used by most of the MainView products.

It manages communication with other CASs on other local and remote systems and allows direct communication between an individual terminal session and a product address space. Usually, one CAS runs per system image. A single CAS can communicate with an unlimited number of remote systems and CASs.

A product establishes an independent connection with its local CAS, so you can add new products or new upgrades to the architecture without affecting existing products or other configurations.

Each CAS contains a component called Plex Manager that provides administration and operations views that help you:

- Manage communication links with other CASs
- Monitor the activity of accessible products
- Create single system image (SSI) contexts
- Control security for products

Product address spaces

The product address space (PAS) runs as a started task.

It comprises special routines, including data collectors, to support one or more MainView products.
### Table 1: Types of PASs

<table>
<thead>
<tr>
<th>PAS type</th>
<th>Products supported</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>z/OS PAS</strong></td>
<td>■ CMF MONITOR ■ MainView for z/OS ■ MainView for UNIX System Services ■ MainView SYSPROG Services ■ MainView VistaPoint (for IBM MVS workloads)</td>
<td>One z/OS PAS runs per system image. The z/OS PAS always connects to the CAS on that system image.</td>
</tr>
<tr>
<td><strong>BBI-SS PAS</strong></td>
<td>■ MainView AutoOPERATOR ■ MainView FOCAL POINT ■ MainView for CICS ■ MainView for DB2 ■ MainView for DBCTL ■ MainView for IMS Online ■ MainView for MQ ■ MainView VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)</td>
<td>Multiple instances of the BBI-SS PAS can run on a single system image and run one or more products. Depending on the products that are installed, the BBI-SS PAS might or might not connect to a CAS on that system image. BBI-SS PASs on local and remote systems are linked together to provide cross-system communication for an individual terminal session.</td>
</tr>
<tr>
<td><strong>product-specific PAS</strong></td>
<td>■ MainView for IP ■ MainView for Linux - Servers ■ MainView for VTAM ■ MainView for WebSphere Application Server ■ MainView Storage Resource Manager (SRM) ■ MainView Transaction Analyzer</td>
<td>For MainView for Linux - Servers, multiple instances of the PAS can run on a single system image. In the case of MainView for Linux - Servers, each PAS can monitor up to 500 Linux images. You can run multiple PASs to support the number of Linux images that you plan to monitor. For MainView for IP, MainView SRM, MainView Transaction Analyzer, and MainView for VTAM, only one product-specific PAS can be active for each product on a system image.</td>
</tr>
</tbody>
</table>
User address spaces

Each terminal session runs in its own user address spaces (UAS). A terminal session provides the end-user session for all MainView products.

The terminal session connects to a CAS (if a CAS is available), to a BBI-SS PAS, or to both.

Table 2: Types of UASs

<table>
<thead>
<tr>
<th>UAS type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MainView Host Gateway using MainView Explorer</td>
<td>Using the MainView Explorer web browser, your terminal session can access MainView products from a personal computer.</td>
</tr>
<tr>
<td>TSO address space</td>
<td>Using a TSO address space, your terminal session can access MainView products and perform other TSO/ISPF functions.</td>
</tr>
<tr>
<td>VTAM or EXCP address space using MainView Alternate Access</td>
<td>Using a separate address space that communicates with your terminal with either VTAM or EXCP, your terminal session can access MainView products and also perform other ISPF functions.</td>
</tr>
</tbody>
</table>

Runtime Component System (RTCS)

RTCS runs as a started task that is started soon after the system is IPLed.

This address space provides programming services to all the CASs, PASs, and UASs. There are no user interfaces other than console commands to modify its operation. RTCS is designed for continuous operation and seldom if ever needs to be stopped.

For information about RTCS, see the BMC Runtime Component System Configuration and Administration Guide.

Administrator tasks

This topic lists the tasks that you need to complete as an administrator based upon the products that are installed at your site.

Use Table 3 on page 18 as a guide through this book.

Each row of the table lists a major task that you are responsible for completing as a MainView administrator. Some tasks must be completed for all MainView products; others need to be completed for specific products only.
**WARNING**

Before you do any of the tasks shown in Table 3 on page 18, review the discussion of MainView product libraries in the *MainView Customization Reference*. The names of distributed product target libraries are used in this manual for reference only. *Distributed libraries should never be modified.* Any changes made to these libraries will be overwritten by subsequent SMP/E maintenance.

Table 3: Administrative tasks for each MainView product

<table>
<thead>
<tr>
<th>Task</th>
<th>Products</th>
<th>Reference</th>
</tr>
</thead>
</table>
| Manage a CAS and cross-system communication | All MainView products except MainView FOCAL POINT  
MainView FOCAL POINT | “Managing the coordinating address space (CAS)” on page 21  
“Managing cross-system communication” on page 97 |
| Manage a z/OS PAS                 | CMF MONITOR  
MainView for z/OS  
MainView for UNIX System Services  
MainView VistaPoint  
MainView SYSPROG Services | “Managing the z/OS product address space (PAS)” on page 37 |
| Manage a BBI-SS PAS               | MainView FOCAL POINT  
MainView AutoOPERATOR  
MainView for CICS  
MainView for DB2  
MainView for DBCTL  
MainView for IMS Online  
MainView for MQ | “Managing the BBI-SS product address space (PAS)” on page 57 |
| Manage MainView Explorer          | All MainView products except MainView FOCAL POINT  
MainView FOCAL POINT | “Managing MainView Explorer” on page 119 |
<table>
<thead>
<tr>
<th>Task</th>
<th>Products</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage MainView Logger</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Managing MainView Logger” on page 127</td>
</tr>
<tr>
<td>Manage targets and systems</td>
<td>All MainView products</td>
<td>“Managing targets and target contexts” on page 149</td>
</tr>
<tr>
<td>Manage single-system image (SSI) contexts</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Managing SSI contexts” on page 183</td>
</tr>
<tr>
<td>Manage historical data sets</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Managing historical data sets” on page 209</td>
</tr>
<tr>
<td>Manage the MainView Infrastructure common registry</td>
<td>All MainView products</td>
<td>“Managing the MainView Infrastructure common registry” on page 231</td>
</tr>
<tr>
<td>Manage the User Interface Middleware (UIM) server</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Managing the User Interface Middleware (UIM) server” on page 235</td>
</tr>
<tr>
<td>Control diagnostic messages</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Controlling diagnostic messages” on page 261</td>
</tr>
<tr>
<td>Display job status information</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Job status information” on page 269</td>
</tr>
<tr>
<td>Create online Help</td>
<td>All MainView products except MainView FOCAL POINT</td>
<td>“Creating online Help” on page 273</td>
</tr>
<tr>
<td>Manage a product-specific PAS</td>
<td>• MainView for IP</td>
<td>topic on managing the PAS in your product documentation</td>
</tr>
<tr>
<td></td>
<td>• MainView for Linux - Servers</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MainView for VTAM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MainView for WebSphere Application Server</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MainView Storage Resource Manager (SRM)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MainView Transaction Analyzer</td>
<td></td>
</tr>
</tbody>
</table>
Managing the coordinating address space (CAS)

The coordinating address space (CAS) runs as a subsystem and is used by most of the MainView products. It manages communication with other CASs on other local and remote systems and provides direct communication between an individual terminal session and a product address space (PAS).

Usually, one CAS runs on a single system image, but there is no limit to the number of remote systems with CASs with which a single CAS can communicate.

This chapter explains how to perform the following tasks:

- “Starting a CAS” on page 21
- “Managing the CASPERM registry” on page 29
- “Stopping a CAS” on page 32
- “Restarting a CAS” on page 34

For information about managing CASs and cross-system communication, see “Managing cross-system communication” on page 97.

Tip

The information described in this chapter applies to all MainView products (except MainView FOCAL POINT).

Starting a CAS

This topic describes how to start a CAS.
To start a CAS

Note

The CAS must be run as a started task. Refer to the *MainView Customization Reference* for instructions on preparing a CAS startup procedure and defining cross-system communication between CASs on multiple systems.

1 Verify that the JCL for the CAS started task procedure exists.

The *MainView Customization Reference* describes the JCL for creating the CAS started task procedure. By default, the CAS started task procedure that is created during customization is called BBMCAS. Your site might have defined a different CAS procedure name and library.

2 From the operator console, enter the START command:

```
S procname[,id][,SSID=ssid][,REUSASID=YES][,parm,parm,...]
```

The table in “CAS START command parameters” on page 22 describes the parameters that can be used with the START command.

3 Look for the following console message that confirms that the CAS was started successfully:

```
BBMZA001I CAS(casename)SSID(ssid)INITIALIZATION COMPLETE - Vv.r.m(modID)
```

CAS START command parameters

The following table describes each parameter that can be used with the START command.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>procname</code></td>
<td>Specifies the name of the startup procedure for the CAS (for example, BBMCAS as created by customization)</td>
</tr>
<tr>
<td><code>.id</code></td>
<td><em>(optional)</em> Assigns an arbitrary ID to the CAS started task. The <code>.id</code> parameter can be used to assign a name to the CAS that is shorter or more recognizable than the procedure name. For example, you might enter START BBMCAS.CASA to start a CAS on system A that is called simply CASA.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| SSID=ssid    | *(required for CAS startup procedure, but optional for START command)*  
Identifies the CAS subsystem ID to the UAS and PAS  
If an SSID is not specified on the START command, the SSID on the PROC statement in the CAS startup procedure is used. The default is BBCS.  
Be sure your MainView users know the SSID value, as this value is required for the **Subsystem ID** field in the Session Control Parameters panel.  
For a PAS to connect to a CAS, the SSID values in both the CAS and PAS startup procedures must be the same. See the *MainView Customization Reference* for more information about creating CAS and PAS startup procedures. |
| REUSASID=YES | *(optional)* For z/OS 1.9 and higher, assigns the address space to the reusable address space pool  
See the START command in z/OS MVS System Commands for details on the usage of this parameter.                                                                                                                                                                                                 |
| CAPS={Y | N}   | *(optional)* Identifies whether the CAS should provide Katakana terminal support for console and WTO messages that are issued from the CAS  
The default is N (No), which means that console messages and WTO messages appear in mixed case. If you initialize the CAS with CAPS=Y, console messages and WTO messages are converted to uppercase for Katakana terminals.                                                                                     |
| COLD={Y | N}   | *(optional)* Indicates whether CAS startup should be a cold start  
The default is N (No).  
When the CAS initializes, several control blocks and load modules are placed in common storage. Most of this storage is freed when the CAS terminates. However, certain blocks are retained to permit the reuse of previously allocated system resources – in particular, system linkage indexes. In addition, two load modules are retained in common storage. COLD=N reuses those control blocks and load modules rather than building new ones.  
Specify COLD=Y only when instructed to do so by BMC Customer Support in an effort to clear an error condition. COLD=Y causes the linkage indexes, control blocks, and load modules to be discarded. The linkage indexes and the common storage that is occupied by those resources are lost until the system is IPLed. Inappropriate use of COLD=Y might exhaust available system linkage indexes, at which point an IPL is required to restore normal system operation. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMORDR={XTV</td>
<td>xxx}</td>
</tr>
<tr>
<td>CONVXCF={Y</td>
<td>N}</td>
</tr>
<tr>
<td>DFLTGRP={BBGROUP</td>
<td>name}</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>DUMP={Y</td>
<td>N</td>
</tr>
<tr>
<td>EMM={Y</td>
<td>N}</td>
</tr>
<tr>
<td>SUB=MSTR</td>
<td><em>(optional)</em> Allows the CAS to start and run independently of JES. <strong>Note:</strong> SUB=MSTR might not be compatible with some releases of OMVS. Early releases of OMVS would not accept a request until their initialization was completed. This effectively disables all TCP/IP communications for the CAS. OMVS releases that are more current queue the request and process it once OMVS is ready. Since this behavior is undocumented and hence subject to change, consider not using SUB=MSTR if also using TCP/IP for communications.</td>
</tr>
<tr>
<td>XDM={Y</td>
<td>N}</td>
</tr>
</tbody>
</table>
Note
The SPCF=Y parameter is still accepted in the CAS startup procedure and on the
START command, however it is ignored. To use the Cross-System Coupling Facility
(XCF) to communicate with other CASs in a sysplex, specify

- CONVXCF=Y
- X on the COMMORDR parameter

Examples of CAS START commands

The following examples show how to start a CAS under various conditions:

- S BBMCAS
  This example uses the default parameters to start a CAS. The BBMCAS parameter
  is the default startup procedure name for the CAS.

- S BBMCAS.TEST,SSID=testid,DFLTGRP=testgrp
  This example uses the default parameters to start a CAS with the identifier of
  TEST. This command can be useful when an existing production CAS is currently
  running on the system. BBMCAS is the startup procedure name for the
  production CAS. BBMCAS.TEST is a unique name for the test CAS. Because
  multiple CASs will be running on the same system, it is necessary to specify a
  unique SSID and XCF group name for the test CAS.

- S BBMCAS,CAPS=Y
  This example starts a CAS that provides Katakana terminal support for console
  and WTO messages.

- S BBMCAS,EMM=Y
  This example starts a CAS with extended message support.

- S BBMCAS,XDM=Y
  This example starts a CAS with extended diagnostic message support. Normally,
  you should use the XDM=Y parameter only under the direction of BMC Customer
  Support to diagnose an error condition.

CAS initialization in a multisystem environment

After a local CAS is started, it attempts to establish cross-system communication
with remote CASs. Specific conditions for the type of communication you are using
must be met before communications can be established between CASs. See one of the following topics for specific conditions:

- “Using XCF communication” on page 27
- “Using TCP/IP communication” on page 28
- “Using VTAM communication” on page 28

When the local CAS initializes (and periodically thereafter), it checks for the availability of each remote CAS that is defined with a CAS definition or that is a member of the same XCF group. After the remote CAS becomes available, the local CAS establishes cross-system communication and issues the following messages:

\[
\text{BBCSB042I Communications initiated with (casName) using ssid} \\
\text{BBMXCL61I Event Based System Monitor to casName initiated using <XCF | TCPIP | VTAM>}
\]

If the attempt to communicate with a remote CAS fails, the following message is issued:

\[
\text{BBCSB043I Unable to communicate with (casName) using ssid}
\]

For more information about configuring CAS definitions for cross-system communication, see “Managing cross-system communication” on page 97. For information about troubleshooting possible cross-system communication problems, see the MainView Customization Reference.

**Using XCF communication**

If you are using XCF to communicate between CASs, the following conditions must be met before communications can be established:

- The Cross-System Coupling Facility (XCF) must be operational.
- All participating CASs must be members of the same sysplex.
- All participating CASs must join the same XCF group. The name of the XCF group can be specified
  - in the XCF Group field of the CAS definitions
  - with the DFLTGRP parameter at CAS startup
  - by default, if no CAS definition is created and DFLTGRP is not specified (BBGROUP if the default group name)
- All participating CASs must include, either explicitly or by default
— the CONVXCF=Y parameter
— X in the COMMORDR parameter

Using TCP/IP communication

If you are using TCP/IP to communicate between CASs, the following conditions must be met before communications can be established:

■ TCP/IP must be available on the local and remote systems.

■ CAS definitions must exist for all participating CASs, and they must include the correct TCP/IP address and port number for each CAS.

■ SUB=MSTR may disable TCP/IP communications for the CAS, as described in the table in “CAS START command parameters” on page 22.

■ Each CAS must have access to an OMVS segment.

■ OMVS and TCP/IP must be completely initialized before the CAS is started.

For more information about using TCP/IP for CAS-to-CAS communication, see “Defining TCP/IP communication” on page 105. For more information about OMVS segment requirements and how to define an OMVS segment in an external security manager (ESM), see “OMVS segment requirements and ESM definitions” on page 331.

Using VTAM communication

If you are using VTAM to communicate between CASs, the following conditions must be met before communications can be established:

■ VTAM must be available on the local and remote systems.

■ CAS definitions must exist for all participating CASs, and they must include the correct VTAM application names for each CAS.

■ The VTAM APPL definition that is specified for each CAS must be active on the system where the CAS is running.

■ A VTAM CDRM definition for the remote CAS must be active on the local system or be reachable by APPN.

Be aware that VTAM can take a long time to initialize after an IPL and might not be available when the local CAS is ready to establish cross-system communication with remote CASs. This situation is particularly true if START commands for VTAM, the CAS, and the PAS have been defined as part of your IPL procedure, such as in SYS1.PARMLIB member COMMNDxx.
You can configure the remote VTAM application to be directly activated at VTAM startup by adding the major node name to SYS1.VTAMLST member ATTCONxx.

Managing the CASPERM registry

The CASPERM registry is a Runtime Component System (RTCS) product registry that stores MainView object definitions (such as dynamic threshold definitions). One CASPERM registry is required for each CAS in your MainView environment; a CASPERM registry cannot be shared by multiple CASs.

Normally, a CASPERM registry data set is allocated for the local CAS during installation, as part of the OZI Customization process. If the CASPERM registry was not created during customization, or you need to create a registry for another CAS in the sysplex, you can use:

- Member BBMCASPR in the BBSAMP sample library
- CASPERM views on the system where you want to create a registry

The following topics describe how to use the CASPERM views to:

- Create a CASPERM registry data set dynamically
- Enable or disable a CASPERM registry
- Change a registry data set name

Note

For information about backing up and restoring the contents of a CASPERM registry, see the discussion of the Registry Maintenance Utility (RMU) in the BMC Runtime Component System Configuration and Administration Guide.

Creating a CASPERM registry data set

Use the following procedure to create a CASPERM registry data set dynamically by using the CASPERMS view.

To create a CASPERM registry

1. Display the CASPERMS view by performing one of the following actions:
In Plex Manager, type **CASPERMS** on the **COMMAND** line and press **Enter**.

From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the CASPERM view:

```
CON * PLEXMGR;CASPERMS
```

In SSI mode, the CASPERMS view displays one line of data for each CAS in the sysplex.

2 Type **CRE** in the line command field next to the CAS for which you want to create a CASPERM registry, and press **Enter**.

**Note**
You can also hyperlink from the **System** field to display the CASPERM view for a specific system. The CASPERM view supports the **CHAnge primary** command for creating or modifying a CASPERM registry.

3 In the Create CASPERM Registry Data Set dialog, specify the following information:

- Data set name for the CASPERM registry, which must be unique to the local CAS
- *(optional)* Symbols to be substituted in the data set name, such as the CAS subsystem ID or z/OS system symbols
- Space, in cylinders, for the registry data set (minimum of 185 cylinders)
- *(optional)* Device type or generic unit
- *(optional)* SMS management class, storage class, and data class values

For more information about completing the Create CASPERM Registry Data Set dialog, see the online Help.

4 Type one of the following commands on the **COMMAND** line and press **Enter**:

- **CREate** to create a new registry data set and save its name in the CASPERM definition
- **SUBstitute** to resolve any data set name symbols and save the resulting name in the CASPERM definition
- **SAVE** to save the name of an existing data set in the CASPERM definition

5 Type **END** on the **COMMAND** line and press **Enter**.
The CASPERMS view is redisplayed, as shown in Figure 2 on page 31.

**Figure 2: CASPERMS view**

```
  ddmmyyyy  hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) ------------
  COMMAND ===>
  CURR WIN ===>
  ALT WIN ===>
  >W1 =CASPERMS==========SJSCC610=*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D====1
  CMD System   Status   Act Cur DS  Dataset
  ---                   Reg Reg Exi Name
  SJSCC610 CREATED  NO  YES YES MV.QA61.SJSC.MV61CAST.CASPERM
```

To display more detailed information about a registry data set, use the System hyperlink field to display the CASPERM view.

---

**Note**

To make a CASPERM registry active and available to store definitions, you must enable it, as described in “Enabling a CASPERM registry” on page 31.

---

### Enabling a CASPERM registry

Use the following procedure to enable a CASPERM registry. A CASPERM registry must be enabled before it can be used to store MainView object definitions.

---

**Note**

If the CASPERM registry data set was configured by using the Installation System, the registry is enabled automatically.

---

**To enable a CASPERM registry**

1. Perform one of the following actions:
   - From the CASPERMS view, type `ENA` in the line command field next to the registry that you want to enable, and press Enter.
   - From the CASPERM view, type `ENAble` on the COMMAND line and press Enter.

### Disabling a CASPERM registry

Use the following procedure to disable a CASPERM registry, which makes it unavailable the next time the CAS is restarted.

**To disable a CASPERM registry**

1. Perform one of the following actions:
■ From the CASPERMS view, type DIS in the line command field next to the registry that you want to disable, and press Enter.

■ From the CASPERM view, type DISable on the COMMAND line and press Enter.

Changing a registry data set name

Use the following procedure to change the name of a CASPERM registry data set.

To change a registry data set name

1 On the CASPERM view, type DISable on the COMMAND line and press Enter to disable the current registry data set.

2 Type CHAnge on the COMMAND line and press Enter.

3 In the Create CASPERM Registry Data Set dialog, specify the name of an existing data set.

4 Press END to save the definition and exit the dialog.

Note
If the data set that you specified does not exist, you cannot save the definition.

After you restart the CAS, you must enable the new registry data set, as described in “Enabling a CASPERM registry” on page 31.

Stopping a CAS

Because the CAS provides necessary services to PASs and user sessions, a running CAS is essential to the operation of MainView products.

Although some functions can be performed without a running CAS, the overall capability and flexibility of the MainView environment is severely diminished when no CAS is running.

However, under certain circumstances you might need to stop a CAS to perform a particular task, such as:

■ Updating the maintenance level of MainView components
■ Shutting down OMVS to apply system maintenance
- Activating TCP/IP communication
- Changing the libraries that are specified in the CAS JCL
- Restoring the MainView or user libraries that are specified in the CAS JCL
- Changing a CAS runtime parameter that cannot be changed dynamically with a view or MODIFY command

These situations, while infrequent, require that you stop and restart the CAS.

When you stop a CAS, local and cross-system communication is disrupted, which can affect the operation of active PASs. While a CAS is unavailable, data that is only available for display in windows mode will not be available. In addition, because the CAS provides services that are used by PASs for data collection, some data collection might not be performed while the CAS is unavailable.

**Note**

When you stop and restart a CAS, BMC recommends that you also stop and restart all PASs that are connected to that CAS. Otherwise, unpredictable results might occur in some MainView products.

**To stop a CAS**

1. Verify the procedure name for the CAS that you want to stop.
2. *(optional)* Stop all active PASs that are supported by the CAS.
3. Issue the STOP command from the operator console:

   ```
P procname
   ```

   Or, if the CAS was started with an ID, issue this command:

   ```
P id | stepname
   ```

   The table in “CAS START command parameters” on page 22 discusses both of these parameters.

4. Look for the following console message that indicates that the CAS has terminated successfully:

   ```
   BBMZA999I SSID(ssid) shutdown complete - CC=0000
   ```
Restarting a CAS

When a CAS is stopped, it leaves behind a small amount of storage in CSA that is used to process late arriving recovery operations. That storage is reused when you perform a warm start of the CAS (for more information about starting a CAS, see “Starting a CAS” on page 21.

Note
If you perform a cold start of the CAS (COLD=Y), the CSA storage is abandoned. You should perform a cold start only when instructed to do so by BMC Customer Support, or if you have problems with a warm start. For more information about the COLD parameter, see the table in “CAS START command parameters” on page 22.

The remainder of this section describes some special considerations to remember when restarting a CAS.

After applying maintenance

The first time that you restart a CAS after applying maintenance, the CAS might determine that a cold start is required to fully implement the new maintenance. If that is the case, the following messages are displayed in the JES joblog during CAS initialization:

BBMZA098I CAS(WGH0) AUTOMATIC COLD START IN PROGRESS FOR THIS STARTUP
BBMZA104I CAS SSID(WGH0) COLD START IN PROGRESS

Note
If a cold start of the CAS is performed (either automatically or because COLD=Y was specified on the START command), you must stop and restart all connected PASs. A PAS cannot successfully reconnect to a CAS that has been cold started.

When you change the CAS libraries

If you modify the CAS JCL to specify new libraries, remember to update the library names in:

- Corresponding PAS and UAS JCL
- MainView CLIST

Make sure that the library names are correctly updated in all these places before you restart the CAS.
Note
The MainView Alternate Access initialization EXEC, BBVINIT, includes the name of the MainView CLIST and its location. If you rename or relocate the CLIST, remember to update the BBVINIT EXEC.
Managing the z/OS product address space (PAS)

The z/OS product address space (PAS) runs as a started task.

*Tip*

The information described in this chapter applies to the following products only:

- CMF MONITOR
- MainView for z/OS
- MainView for UNIX System Services
- MainView SYSPROG Services
- MainView VistaPoint (for IBM MVS workloads)

During customization, you can define the z/OS PAS so that it is started at IPL (see the MainView Customization Reference for more information). However, if you need to start, modify, or stop the PAS manually, or if you are customizing the PAS for the first time, you might want to refer to information in this chapter.

This chapter begins with a discussion of some questions that are commonly asked by new z/OS PAS users; provides instructions for starting, stopping, and modifying the z/OS PAS; describes each parameter that is defined to the z/OS PAS PROC statement; and discusses how to run two CMF MONITOR Extractors concurrently on the same system for testing purposes.

Understanding z/OS PAS operations

The questions and answers in this topic address some of the questions that many new MainView users have.
What happens when the z/OS PAS is initialized?

First, the z/OS PAS tries to connect to the CAS. Then, if the CAS is available, the z/OS PAS brings up its data collectors, including the CMF MONITOR Extractor, as soon as a connection is made. If the CAS is not available, the z/OS PAS starts the CMF MONITOR Extractor only, and then tries 60 times to establish connection with the CAS, once every 5 seconds. If a connection is made during this time period, the other data collectors are started. If a connection is not made, you must restart the data collectors when the CAS is available by using a MODIFY command with DC=START specified (see “Controlling z/OS PAS operation” on page 52 for more information).

What happens if the CAS becomes unavailable while MainView for z/OS or CMF MONITOR Online is running?

All data collectors terminate except the CMF MONITOR Extractor; however, if you have the PGDDLAY Extractor control statement defined, this sampler goes into a wait state until the CAS returns and the data collectors are reinitialized. After the CAS is back up, to initialize any terminated data collectors and reconnect the PAS to the CAS, you can enter a MODIFY command for the PAS with DC=START specified. You do not have to bring the PAS down and then restart it again – you can reconnect the PAS dynamically and reinitialize the data collectors. The PGDDLAY sampler resumes operation, as well. At that time, you can use the MainView Selection Menu to re-attempt access to your MainView product.

What happens to the Extractor if the CAS or one of the z/OS PAS data collectors goes down?

Nothing. With the exception of the PGDDLAY extractor, after the z/OS PAS is initialized, the CMF MONITOR Extractor remains available at all times, regardless of the status of the CAS or other data collectors in the z/OS PAS.

Can I run two copies of CMF MONITOR on the same system?

Yes. Use one of the following methods:

- Run two z/OS PASs on the same system. For more information about running multiple z/OS PASs, see “Running z/OS PASs concurrently” on page 52.

- Run two Extractors in the same z/OS PAS by using the two CMF MONITOR modes, CPM and IPM. For more information about CPM and IPM modes, see the CMF MONITOR User Guide and Reference.
What if I set things up so the Extractor that is used by the z/OS PAS also records SMF data? Is this data at risk?

Not as long as the z/OS PAS itself is up and running. Even if the CAS or one of the other data collectors goes down, the Extractor continues to write SMF records.

Starting the z/OS PAS

Use this procedure to start the z/OS PAS.

Note
If you are running the Resource Management Facility (RMF) from IBM in addition to MainView for z/OS and CMF MONITOR, proceed as follows:

- You must start RMF before starting the CAS and z/OS PAS for these MainView products.
- You must stop the CAS and z/OS PAS before stopping RMF when you shut down your system.

To start the z/OS PAS

1. Verify that the JCL that is used to start the z/OS PAS has been created (see the MainView Customization Reference for more information).

2. From the operator console, enter the following START command:

   START proname [.id],SYSID=sysid[,XDM={Y | N}][,CXEN={Y | N}][,DC={CPM | IPM | START | STOP}][,CPM={xx | 00}][,IPM={xx | 00}][,EM={xx | 00}][,SSID=ssid] [,CMDID=symbol][,REUSASID=YES]

   The table in “z/OS PAS started task parameters” on page 39 describes each parameter.

3. Look for the following console message that verifies a successful z/OS PAS initialization:

   MVS PAS (ssid) initialization complete, rx.x.x (rmid)

z/OS PAS started task parameters

The following table describes each parameter in the z/OS PAS started task procedure.
### Table 5: z/OS PAS START command parameters

<table>
<thead>
<tr>
<th>PROC Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>.id</code></td>
<td><em>(optional)</em> Assigns an arbitrary ID to the PAS started task. For example, you might enter <code>START MVSPAS.PASA</code> to start a PAS called PASA and <code>START MVSPAS,PASB</code> to start a PAS called PASB. This way, you can control the two PASs independently.</td>
</tr>
<tr>
<td><strong>CMDID=symbol</strong></td>
<td><em>(optional)</em> Defines a single character that can be used in place of the MODIFY command and <code>procname</code> when a MODIFY command is issued to the PAS. You can use the CMDID parameter as a quick method for issuing commands. You must manually add the CMDID parameter to the PAS procedure statement to enable this support. Valid operands are as follows:</td>
</tr>
<tr>
<td>€</td>
<td>(</td>
</tr>
<tr>
<td></td>
<td>&amp;</td>
</tr>
<tr>
<td>)</td>
<td>É</td>
</tr>
<tr>
<td>-</td>
<td>/</td>
</tr>
<tr>
<td>&gt;</td>
<td>?</td>
</tr>
<tr>
<td>=</td>
<td>&lt;</td>
</tr>
<tr>
<td>Certain operands might need to be enclosed within quotation marks to work in your specific environment. Alternatively, operands may be specified as a two-digit hexadecimal representation. For information about using the MODIFY command, see “Controlling z/OS PAS operation” on page 52.</td>
<td></td>
</tr>
<tr>
<td>**CPM={xx</td>
<td>00}**</td>
</tr>
<tr>
<td>PROC Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| CXEN={Y | N}   | *(optional, CMF MONITOR Online only)*  
  Controls whether more than one copy of the Extractor can initialize on  
  the same system  
  The default is Y (Yes), which allows enqueue and workload sampling  
  and writing to SMF data sets.  
  You can run more than one CMF Extractor on your system at a time.  
  However, only one Extractor can sample workload data or enqueue  
  data.  
  By specifying CXEN=N for any additional Extractors, more than one  
  CMF MONITOR Extractor can execute on the same system  
  concurrently. CXEN=N prevents the Extractor from writing to SMF data  
  sets, sampling workload data, and sampling enqueue data.  
  You should not run multiple Extractors because of associated overhead,  
  unless it is required in trial situations to evaluate MainView for z/OS.  
  CXEN=N should be defined to a trial Extractor, if necessary. |
### PROC Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC={CPM</td>
<td>IPM</td>
</tr>
<tr>
<td>START (the default): z/OS PAS data collectors initialize at startup under CPM monitoring mode. When the data collectors are initialized, access to CMF MONITOR Online and MainView for z/OS is provided. The CMF samplers associated with the PGDDLAY and CFDATA Extractor control statements are enabled for sampling.</td>
<td></td>
</tr>
<tr>
<td>STOP: Data collectors do not initialize when the z/OS PAS initializes. You cannot access CMF MONITOR Online or MainView for z/OS. The CMF MONITOR PGDDLAY and CFDATA Extractor control statements do not sample data because both gather information from the z/OS PAS data collectors.</td>
<td></td>
</tr>
<tr>
<td>CPM: z/OS PAS data collectors initialize and accept data from the Extractor in CPM mode at startup. Specifying DC=CPM is the same as DC=START.</td>
<td></td>
</tr>
<tr>
<td>IPM: z/OS PAS data collectors initialize and accept data coming from the Extractor in IPM mode at startup.</td>
<td></td>
</tr>
</tbody>
</table>

If you specify DC=IPM, you must also specify the parameter IPM=xx, where xx is the suffix of the CMFIPMxx control statement member that is pointed to by the //PARMLIB DD statement.

The DC parameter can also be changed by using a MODIFY command, so the data collectors can be started or stopped or the monitoring modes switched without requiring the z/OS PAS (and Extractor) to be stopped and then started. You can specify an additional operand with the DC parameter when the MODIFY command is issued:

**STATUS** (the status of each data collector is displayed in the job log)

**WARNING:** DC=STATUS should not be specified at z/OS PAS initialization because the data collectors do not initialize. It should be used only when the PAS is running.

For more information about how to use the MODIFY command, see “Controlling z/OS PAS operation” on page 52.

### EM={xx | 00} |

*(MainView for z/OS only)*

Specifies the two-character suffix of the PWSCPMxx control statement member that you want the MainView for z/OS Exception Monitor to use.
### PROC Parameter Description

<table>
<thead>
<tr>
<th>PROC Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPM={xx</td>
<td>00}</td>
</tr>
<tr>
<td>procname</td>
<td>Specifies the name of the startup procedure for the PAS (for example, MVSPAS)</td>
</tr>
<tr>
<td>REUSASID=YES</td>
<td><em>(optional)</em> For z/OS 1.9 and higher, assigns the address space to the reusable address space pool. See the START command in the IBM book <em>z/OS MVS System Commands</em> for details on the usage of this parameter.</td>
</tr>
<tr>
<td>SSID=ssid</td>
<td>Specifies the subsystem ID of the CAS to which the z/OS PAS should connect. The CAS is started as a separate address space from the z/OS PAS and must be initialized before the z/OS PAS is initialized. For a PAS to connect to a CAS, the SSID values in both the CAS and PAS startup procedures must be the same.</td>
</tr>
<tr>
<td>SYSID=sysid</td>
<td>Specifies the SMF system ID for your system</td>
</tr>
<tr>
<td>USEZIIP=(Y</td>
<td>N)</td>
</tr>
<tr>
<td>PROC Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>XDM={Y</td>
<td>N}</td>
</tr>
</tbody>
</table>
PROC Parameter | Description
--- | ---
XDS={xx|STOP} | *(optional, CMF MONITOR Online only)*

Controls the collection of data by the cross-system data server (XDS)

To share cross-system data, all the systems must be in the same sysplex, the z/OS PAS on each system must have XDS active, DC=START must be specified for each z/OS PAS, and the CAS on all systems must be in the same CASDEF XCF group.

For SDSF to use XDS data, the BBLINK data set needs to be either in the link list or in the TSO logon procedure STEPLIB.

To collect XDS data, specify XDS=xx, where xx indicates the suffix of a CMFXDSxx member of hilevel.UBBPARM that was created by customization. You can switch to a different XDS member without stopping either the z/OS PAS or XDS by using the MODIFY command. For example, if you are running the z/OS PAS that points to the CMFXDS00 member with XDS=00, you can begin using the CMFXDS01 member by issuing the MODIFY command

```
F MVSPAS,XDS=01
```

You can also stop XDS without stopping the z/OS PAS by issuing the MODIFY command

```
F MVSPAS,XDS=STOP
```

The three parameters for the CMFXDSxx members are TYPE, RECORDS, and SIZE. Each of these parameters may be specified only once in a particular XDS member. If a line begins with an asterisk (*), that line is ignored. Data in columns 73-80 is also ignored.

TYPE defines which SMF record types are included in the XDS SMF record buffer for the CX10XDQY and CX10XDRC APIs. Specify one of the following values:

- Specify **TYPE CMF** to use output records from CMF itself. Use this parameter to buffer all type 70-78 records, as well as all CMF user record types. This option is the default value if the TYPE parameter is not specified.

- Specify **TYPE SMF** to use SMF IEFU83, IEFU84, and IEFU85 dynamic exits. Use this parameter to buffer all record types except those written to CMF Extractor output data sets. If the CMF Extractor is writing to SMF, TYPE SMF works exactly the same as TYPE ALL.

- Specify **TYPE ALL** to use both SMF-provided and CMF-provided records.

- Specify **TYPE NONE** to allow API calls and CX10XDGS requests for type 79 data from this system without buffering records on this system.
<table>
<thead>
<tr>
<th>PROC Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| RECORDS={xx|STOP} (continued) | RECORDS defines which SMF record types and subtypes are included in the XDS buffer. This parameter must specify a subset of the records that are specified for collection by the TYPE parameter. Records that are specified with this parameter are available for CX10XDQY (XDS record query) and CX10XDRD (XDS record retrieval) API calls. In addition, cross-system snapshots of type 79 data are available from the CX10XDGS (XDS data-gathering service) API call. Specify the records that you want to include in the buffer by listing those types in a list (which may continue on multiple lines) that follows the RECORDS parameter. For example, if you want to buffer type 70 and 72 records only, you specify RECORDS 70,72. You can specify subtypes by placing a hyphen (-) after a type, followed immediately by the subtype. For example, to buffer only subtype 3 of type 72 records, specify RECORDS 72-3. Ranges of types and subtypes can be specified either by using a colon to separate the start and end of a range or by using an X character to indicate all digits in a range. For example, to buffer record types 70 through 79, you can specify either RECORDS 70:79 or RECORDS 7X. If you do not want to buffer any records, specify RECORDS NONE. If you want to buffer all record types included in the TYPE parameter, specify RECORDS ALL. This option is the default if you do not include a RECORDS parameter. SIZE defines how much SMF data should be buffered before records are lost because of buffer wraparound. Specify a size in the format of SIZE nnnu, where nnn is a one- to nine-digit number and u is one of the following values: 
- blank (bytes)
- K (kilobytes) (2**10 bytes)
- M (megabytes) (2**20 bytes)
- G (gigabytes) (2**30 bytes)
For example, for a buffer size of 3 megabytes, specify SIZE 3M. If you do not specify the SIZE parameter, the default buffer size is 32M. You can also specify the word PURGE with this parameter to indicate that the existing buffer should be discarded at the time of a MODIFY command, rather than when the new buffer fills up. For example, to change a buffer size from 3 megabytes to 5 megabytes and immediately discard the 3 megabyte buffer, specify SIZE 5M,PURGE on the member referred to by the MODIFY command. |
| | |
Examples of z/OS PAS START commands

The following examples show various ways that the START command can be used to start the z/OS PAS:

■ S MVSPAS,SYSID=SYSA
   This example uses the default parameters to start an z/OS PAS, where MVSPAS is the startup procedure name for the PAS and SYSA is the SMF system ID.

■ S MVSPAS,SYSID=SYSA,DC=STOP
   This example starts the Extractor in the z/OS PAS, but not the data collectors. The DC=STOP parameter prohibits the data collectors from initializing, which also means the PGDDLAY and CFDATA control statements for the Extractor have no data to collect.

■ S MVSPAS.TEST,SYSID=SYSA,CXEN=N
   This example starts an alternate z/OS PAS when a primary z/OS PAS is already running. MVSPAS is the startup procedure name for the PAS, TEST is a unique .id value, and SYSA is the SMF system ID. The CXEN=N parameter allows an alternate Extractor to initialize on a system where a primary Extractor is already running.

■ S MVSPAS,SYSID=SYSB,EM=01
   This example starts an z/OS PAS with a different control statement member for the MainView for z/OS Exception Monitor. The EM=01 parameter initializes the Exception Monitor by using the control statements in BBPARM member PWSCPM01.

■ S MVSPAS,SYSID=SYSC,XDM=Y
   This example modifies an z/OS PAS when BMC Customer Support has requested that you enable extended diagnostic message (XDM) mode to help determine the cause of an error condition.
   
   Note
   Do not enable the XDM parameter unless requested to do so by Customer Support.

■ S MVSPAS,SYSID=SYSB,CPM=04,IPM=01
   This example starts an z/OS PAS with different IPM and CPM control statement members for the Extractor. The CPM=04 parameter initializes the CPM monitoring mode of the Extractor by using the control statements in BBPARM member CMFCPM04. The IPM=01 parameter initializes the IPM monitoring mode of the Extractor by using the control statements in BBPARM member CMFIPM01.
About CPM control statement members

The z/OS PAS startup procedure contains a CPM parameter that points to a control statement member that defines the operating conditions and data you want the CMF MONITOR Extractor and the system data collectors to use for CPM monitoring mode.

The default control statement member is CMFCPM00.

To use a control statement member other than CMFCPM00

1. Create a member by the name of CMFCPMxx, where xx is a unique two-character alphanumeric suffix in hilevel.UBBPARM (as created by customization).

2. Specify the unique two-character suffix on the CPM parameter and enter a MODIFY command that points to the new member (see “Controlling z/OS PAS operation” on page 52) or restart the z/OS PAS.

3. To define the new member so that it is read at z/OS PAS initialization, do one of the following actions:
   - Copy the control statement member into the data set allocated by the DD name PARMLIB (hilevel.UBBPARM by default).
   - If you have an existing data set that contains the control statement member, concatenate that data set name to the data set allocated by the DD name PARMLIB. Your concatenation should look something like this example:

     ```
     //PARMLIB DD DISP=SHR,DSN=hilevel.CMFdsn
     // DD DISP=SHR,DSN=hilevel.UBBPARM
     // DD DISP=SHR,DSN=hilevel.BBPARM
     ```

     where CMFdsn is a data set containing the desired CPM control statement member.

The CMFCPM00 default member was constructed during customization (see the MainView Customization Reference for more information). The control statements that are contained in CMFCPM00 vary, depending on the combination of products that you have installed.

For example, if you have MainView for z/OS, CMF MONITOR, or both installed, CMFCPM00 looks like Figure 3 on page 48.

Figure 3: CMFCPM00 for MainView for z/OS and CMF MONITOR

```
***********************************************************************
*                                                                     *
*      SAMPLE CMF EXTRACTOR CONTROL CARDS FOR USE WITH THE             *
*      MAINVIEW for z/OS.                                             *
*                                                                     *
*      THE SAMPLERS SPECIFIED BELOW REPRESENT THE MINIMUM SET          *
*      REQUIRED BY MAINVIEW for z/OS DATA COLLECTORS.                 *
***********************************************************************
```

Starting the z/OS PAS

48 MainView Administration Guide
ADDITIONAL SAMPLER CONTROL STATEMENTS MAY BE ADDED AS NEEDED TO SATISFY OTHER REPORTING REQUIREMENTS.

REPORT CPM, INTERVAL=QTR, RUNTIME=1440, SMFRECID=240, GBLS=1000, CSA=100, SMF=NO, SYNCH=00

ASMDATA

<table>
<thead>
<tr>
<th>CHANNEL</th>
<th>SAMPLE=500</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>CPU</td>
</tr>
<tr>
<td>DEVICE</td>
<td>SAMPLE=500, CLASS=DASD</td>
</tr>
<tr>
<td>DEVICE</td>
<td>SAMPLE=1000, CLASS=TAPE, OFFLINE=YES</td>
</tr>
<tr>
<td>*ENQUEUE</td>
<td></td>
</tr>
<tr>
<td>PAGING</td>
<td>SAMPLE=5000</td>
</tr>
</tbody>
</table>

The statements shown in Figure 3 on page 48 are described in Table 6 on page 49.

Table 6: Description of Extractor statements in sample CMFCPM00 member

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT</td>
<td>Sets global parameters for an Extractor run</td>
</tr>
<tr>
<td></td>
<td>■ CPM specifies continuous (as opposed to intermittent) monitoring mode.</td>
</tr>
<tr>
<td></td>
<td>■ INTERVAL=QTR specifies that data from the samplers is gathered every 15 minutes, on the quarter-hour.</td>
</tr>
<tr>
<td></td>
<td>■ RUNTIME=1440 specifies the maximum number of minutes that CPM mode remains active.</td>
</tr>
<tr>
<td></td>
<td>■ SMFRECID=240 specifies an identification for SMF records (if you choose to have the z/OS PAS CMF Extractor write SMF records).</td>
</tr>
<tr>
<td></td>
<td>When started with this version of CMFCPM00, the Extractor writes SMF data to the data sets that are specified on the CMFCPM1 and CMFCPM2 DD statements.</td>
</tr>
<tr>
<td></td>
<td>■ GBLS=1000 specifies that the global sampler should gather data every 1000 milliseconds.</td>
</tr>
<tr>
<td></td>
<td>■ CSA=100 specifies the amount of CSA required, 100K.</td>
</tr>
<tr>
<td></td>
<td>■ SMF=NO specifies that this control statement member does not write SMF records.</td>
</tr>
<tr>
<td></td>
<td>■ SYNCH=00 specifies that the recording interval is not synchronized.</td>
</tr>
<tr>
<td>ASMDATA</td>
<td>Collects auxiliary storage management data, including information about the I/O activity of page and swap data sets and Auxiliary Storage Manager (ASM) data constants</td>
</tr>
<tr>
<td></td>
<td>Because a value is not specified, the default value is used; data is collected at a rate of once per second.</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>CHANNEL</td>
<td>Collects channel path statistics from the channel path measurement table at the beginning and end of each recording interval</td>
</tr>
<tr>
<td>CPU</td>
<td>Collects CPU and CPU-dispatching data, including information about CPU wait, busy, and idle status, online and offline times, queue depth, and processor concurrency. SAMPLE=500 means CPU data is collected every 500 milliseconds (or every half second).</td>
</tr>
<tr>
<td>DEVICE</td>
<td>Measures DASD devices for busy and wait status, I/O activity, online and offline times, and volume activity. SAMPLE=500 indicates DASD devices are measured every 500 milliseconds (or every half second). CLASS=DASD indicates the device type as DASD devices.</td>
</tr>
<tr>
<td>DEVICE</td>
<td>Measures tape devices for busy and wait status, I/O activity, online and offline times, and volume activity. SAMPLE=1000 indicates DASD devices are measured once per second. CLASS=TAPE indicates the device type as tape devices. OFFLINE=YES indicates that offline devices are sampled.</td>
</tr>
<tr>
<td>ENQUEUE</td>
<td>Usually collects activity on queuing conflicts. In CMFCPM00, however, this statement is commented out to allow you to run two copies of the CMF Extractor simultaneously. (Only one concurrently active copy of the CMF Extractor can monitor enqueue activity. This restriction does not affect your MainView for z/OS or CMF MONITOR data.) If you decide to use the z/OS PAS Extractor for production, be sure to remove the comment from this statement.</td>
</tr>
<tr>
<td>PAGING</td>
<td>Collects paging and swapping data. SAMPLE=5000 means paging and swapping data is sampled every 5 seconds.</td>
</tr>
</tbody>
</table>

**About IPM control statement members**

The z/OS PAS startup procedure contains an IPM parameter that points to a control statement member that defines the operating conditions and data that you want the CMF MONITOR Extractor and the system data collectors to use for IPM monitoring mode.

The default in the startup procedure is IPM=STOP, which specifies that no IPM member is used and the Extractor is not initialized to run in IPM mode.

The default CMFIPM00 member was constructed during customization (see the MainView Customization Reference for more information). The control statements that are contained in CMFIPM00 vary, depending on the combination of products that you have installed.
To use a control statement member other than CMFIPM00

1. Create a member by the name of CMFIPMxx, where xx is a unique two-character alphanumeric suffix in hilevel.UBBPARM (as created by customization).

2. Specify the unique two-character suffix on the IPM parameter and enter a MODIFY command that points to the new member (see “Controlling z/OS PAS operation” on page 52) or restart the z/OS PAS.

3. To define the new member so that it is read at z/OS PAS initialization, perform one of the following actions:
   - Copy the control statement member into the data set allocated by the ddname PARMLIB (hilevel.UBBPARM by default).
   - If you have an existing data set that contains the control statement member, concatenate that data set name to the data set allocated by the ddname PARMLIB. Your concatenation should look like this example:

     ```
     //PARMLIB DD DISP=SHR,DSN=hilevel.CMFdsn
     // DD DISP=SHR,DSN=hilevel.UBBPARM
     // DD DISP=SHR,DSN=hilevel.BBPARM
     ```

     where CMFdsn is a data set that contains the desired IPM control statement member.

For example, if you have MainView for z/OS or CMF MONITOR Online installed, CMFIPM00 looks like Figure 4 on page 51.

**Figure 4: CMFIPM00 for MainView for z/OS and CMF MONITOR**

```
* ******************************************************
* SAMPLE CMF EXTRACTOR CONTROL CARDS FOR USE WITH THE  *
* MAINVIEW for z/OS.                                   *
* THE SAMPLERS SPECIFIED BELOW REPRESENT THE MINIMUM SET *
* REQUIRED BY MAINVIEW for z/OS DATA COLLECTORS.        *
* ADDITIONAL SAMPLER CONTROL STATEMENTS MAY BE ADDED AS *
* NEEDED TO SATISFY OTHER REPORTING REQUIRMENTS.        *
* ******************************************************
* REPORT IPM,INTERVAL=2,RUNTIME=1440,SMFRECID=240,     *
*   GBLS=NO,CSA=150,SMF=NO                             *
* ASMDATA                                              *
* CHANNEL                                              *
*   CPU       SAMPLE=500                               *
*   DEVICE    SAMPLE=500,CLASS=DASD                    *
*   DEVICE    SAMPLE=1000,CLASS=TAPE,OFFLINE=YES       *
*   ENQUEUE   SAMPLE=5000                              *
*   Paging    SAMPLE=5000                              *
```

See the table in “About CPM control statement members” on page 48 for information about the sampler control statements that are shown in Figure 4 on page 51. For more information about IPM mode and the sampler control statements, see the CMF MONITOR User Guide and Reference.
Running z/OS PASs concurrently

The z/OS PAS contains the CMF MONITOR Extractor.

Any number of extractors can collect enqueue data in a single system image. However, only one extractor can write SMF records at a time in a single system image.

If you already have the following resources in place, the CXEN parameter (see “z/OS PAS started task parameters” on page 39 on page 48) can be used to run two z/OS PASs or an Extractor and an z/OS PAS concurrently:

- Production z/OS PAS that is running on a system where you want to test another z/OS PAS
- Existing Extractor that is used by another product, such as DASD ADVISOR

To run a test z/OS PAS on the same system as a production z/OS PAS (or Extractor-based product), specify CXEN=N in the test z/OS PAS started task procedure before PAS initialization.

**Note**
You should replace your production Extractor or z/OS PAS with the test z/OS PAS as soon as you feel comfortable doing so.

Controlling z/OS PAS operation

You might need to modify the operation of the z/OS PAS or the CMF MONITOR Extractor while it is running, such as start or stop the data collectors, change the Extractor monitoring mode, or change the configuration of your Extractor control statement set.

MODIFY commands

Using a MODIFY command, you can change some of the z/OS PAS parameters without having to recycle the PAS or disrupt Extractor recording.

The following parameters can be used in a MODIFY command:

- CPM
- DC
Controlling z/OS PAS operation

- IPM
- R
- STDC
- USEZIIP
- XDS

**Note**

The R parameter is available only if you have MainView for z/OS installed. With this parameter, you can enter a MODIFY command that executes a SYSPROG service from the PAS. The operand for the R parameter is any valid SYSPROG command name. For more information about SYSPROG commands, see the *MainView for z/OS User Guide and Reference*. The remaining PAS parameters can be used with a START command only.

Table 7 on page 53 shows examples of valid MODIFY commands that can be used to control the z/OS PAS and the Extractor.

**Table 7: MODIFY command examples for the z/OS PAS**

<table>
<thead>
<tr>
<th>If you enter this value</th>
<th>z/OS PAS performs this action</th>
</tr>
</thead>
<tbody>
<tr>
<td>F MVSPAS,CPM=07</td>
<td>Invokes CPM monitoring mode with control statement member CMFCPM07 when the PAS is initialized with CMDID=&gt;</td>
</tr>
<tr>
<td>F MVSPAS,IPM=XY</td>
<td>Invokes IPM monitoring mode with control statement member CMFIIPMXY when the PAS was initialized with CMDID=&gt;</td>
</tr>
<tr>
<td>F MVSPAS,CPM=STOP</td>
<td>Stops CPM monitoring mode. The z/OS PAS is also stopped, unless IPM monitoring mode is active. If DC=START was specified and IPM mode is active, the data collectors switch to DC=IPM. If DC=CPM was specified, the data collectors are stopped, regardless of the status of IPM mode.</td>
</tr>
<tr>
<td>F MVSPAS,IPM=STOP</td>
<td>Stops IPM monitoring mode. CPM monitoring mode remains active. If DC=START was specified, the data collectors switch to DC=CPM. If DC=IPM was specified, the data collectors are stopped.</td>
</tr>
<tr>
<td>F MVSPAS,DC=STOP</td>
<td>Stops the data collectors. CMF MONITOR Online and MainView for z/OS are unavailable. For CMF MONITOR customers, the samplers for the PGDDLAY and CFDATA control statements, if defined, suspend their sampling functions.</td>
</tr>
<tr>
<td>F MVSPAS,DC=START</td>
<td>Invokes the data collectors. CMF MONITOR Online and MainView for z/OS are initialized. For CMF MONITOR, the samplers for the PGDDLAY and CFDATA control statements, if defined, begin or resume their sampling functions.</td>
</tr>
</tbody>
</table>
If you enter this value | z/OS PAS performs this action
--- | ---
F MVSPAS,DC=STATUS | Produces a status display of the z/OS PAS data collectors on the console
F MVSPAS,STDC=START | Starts short-term data collection by using BBPARM member BBDTST00
Short-term data collection is introduced in MainView for z/OS version 3.0.00 and CMF MONITOR version 5.8.00.
**Note:** The data collectors must be active. See the DC parameter examples.
F MVSPAS,STDC=15 | Starts short-term data collection by using BBPARM member BBDTST15
**Note:** The data collectors must be active. See the DC parameter examples.
F MVSPAS,STDC=STOP | Stops short-term data collection
F MVSPAS,XDS=00 | Starts (or restarts) XDS by using BBPARM member CMFXDS00
F MVSPAS,USEZIIP=N | CPU processing that is eligible to be offloaded to zIIPs is now execute on standard CPs

For MainView for z/OS, the R=sysprog parameter provides the ability to invoke SYSPROG services from the system console. The sysprog value includes the name of the service and any parameters that are required by the service. If the CMDID=symbol parameter is also defined, you can invoke the SYSPROG services from the system console as follows:

```
symbolsymbolsysprog
```

where

- **symbol** is a previously defined CMDID character.
- **sysprog** is a SYSPROG service name.

For example, if CMDID=# is defined in the z/OS PAS started task procedure, you can invoke the SYSPROG INFO service by entering ##INFO from the system console. One # symbol replaces the F MVSPAS value and the other # symbol replaces the R parameter.

## Stopping the z/OS PAS

If you stop the z/OS PAS, any local or cross-system sessions with CMF MONITOR Online or MainView for z/OS are disrupted.
BMC recommends the following actions for CMF MONITOR to adjust for Daylight Saving Time changes:

- **For Spring Daylight Saving Time**, stop the z/OS PAS prior to the time change and start the PAS again after the time change.

- **For Autumn Daylight Saving Time**, stop the z/OS PAS prior to the time change and wait one hour before restarting.

Although these changes are meant specifically to accommodate CMF MONITOR SMF records (which use local time), be aware that stopping the PAS affects not only CMF MONITOR but also other products that are running in the PAS, such as MainView for z/OS, MainView for UNIX System Services, or MainView SYSPROG Services.

By using the CONtext command to display the DIAGSESS view in PLEXMGR, you can see the active sessions on the system that you are viewing.

### To stop the z/OS PAS

1. Verify the procedure name for the z/OS PAS that you want to stop.

2. From the system operator console, enter the STOP command:

   ```
P  procnane
   ```

   Or, if the PAS was started with an ID, enter

   ```
P  id
   ```

**Note**
BMC recommends using the STOP command rather than the CANCEL command to stop the z/OS PAS. “z/OS PAS started task parameters” on page 39 discusses these two parameters.

Optionally, you can stop the z/OS PAS data collectors only and leave just the Extractor operating to conserve system resources; for example, when you are not using CMF MONITOR Online. When you stop the data collectors, any local or cross-system sessions with CMF MONITOR Online or MainView for z/OS are disrupted, but the Extractor is not.

To enable and disable the z/OS PAS data collectors without affecting the Extractor, see information about the DC parameter in “Controlling z/OS PAS operation” on page 52.
Managing the BBI-SS product address space (PAS)

This chapter discusses the operation of the BBI-SS PAS. It includes procedures to start, modify, and stop the BBI-SS PAS, as well as controlling BBI-SS PAS resources.

Tip

The information described in this chapter applies to the following products only:

- MainView AutoOPERATOR
- MainView FOCAL POINT
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for MQ
- MainView VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)

Starting a BBI-SS PAS

BBSAMP member SSJCL contains sample JCL to start a BBI-SS PAS.

You can edit this sample JCL to create a job or a procedure and to identify the type of start.

If the PROC statement is used, the BBI-SS PAS is activated as a started task. If the JOB statement is used, the BBI-SS PAS is activated by submitting the job. The BBLINK library must be authorized or initialization will fail.
The default region size is 64 MB to enable product access to sufficient extended storage. Storage is allocated only when needed.

Because the BBI-SS PAS is independent of the target system, the BBI-SS PAS should be started at IPL and remain active regardless of the status of the target systems.

To start the BBI-SS PAS, issue the following START command:

\[ S \text{ procname}[.id][\text{SSID}=	ext{ssid}][,\text{START}=	ext{type}] \]

where the components of the command are as follows:

<table>
<thead>
<tr>
<th>Command component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>procname</td>
<td>Specifies the name of the startup procedure for the PAS</td>
</tr>
<tr>
<td>.id</td>
<td>(optional) assigns an arbitrary ID to the PAS started task</td>
</tr>
<tr>
<td>SSID=ssid</td>
<td>(optional) Specifies the subsystem ID for your system</td>
</tr>
<tr>
<td></td>
<td>This parameter overrides the subsystem ID specified in the PROC member</td>
</tr>
<tr>
<td>START=type</td>
<td>(optional) Specifies the type of BBI-SS PAS startup where type can be one of the following values:</td>
</tr>
<tr>
<td></td>
<td>■ WARM reuses control blocks previously obtained in CSA (default).</td>
</tr>
<tr>
<td></td>
<td>■ COLD frees control blocks held by any previous BBI-SS PAS and obtains new control blocks.</td>
</tr>
<tr>
<td></td>
<td>■ FREE frees control blocks in the same manner as a cold start and terminates the BBI-SS PAS immediately after the storage is freed.</td>
</tr>
<tr>
<td></td>
<td>■ KILL removes the BBI-SS PAS from the system as if it was never started and leaves control blocks allocated.</td>
</tr>
</tbody>
</table>

**Note:** The KILL option should only be used in an emergency when the PAS cannot be restarted by any other method. The next time the BBI-SS PAS is started, an automatic cold start is performed.

A warm start is the default and should be used whenever possible. See “Warm start of a BBI-SS PAS” on page 59 and “Cold start of a BBI-SS PAS” on page 59 for more information.

If, during a BBI-SS PAS start, a link to another BBI-SS PAS fails because VTAM cannot complete the connection, the following message is issued:

PM0503W BBI CONNECTION FAILED FROM subsys TO subsys
where subsys is the one- to four-character BBI-SS PAS ID.

After this message is issued, BBI initialization continues. When the problem with the link is corrected, the link can be started with the BBI control command START LINK (see “BBI control commands” on page 77 for more information about the START command).

### Warm start of a BBI-SS PAS

A warm start reuses control blocks previously obtained in CSA.

Terminal session users that were connected to the BBI-SS PAS remain connected within their current applications. If an application was in refresh mode, it continues to refresh.

A warm start should be used whenever possible. By default, the JCL in BBSAMP member SSJCL specifies START=WARM.

### Cold start of a BBI-SS PAS

A BBI-SS PAS cold start frees control blocks held by any previous BBI-SS PAS and obtains new control blocks for the new BBI-SS PAS.

It can be automatic or requested. A cold start request is specified in BBSAMP member SSJCL as START=COLD.

A cold start might be necessary whenever PRODUCT statements are added, removed, or modified in BBPARM member BBISSP00. Otherwise, this request should be made only because BMC Customer Support has recommended a cold start or because it is the only way to start a BBI-SS PAS.

**Note**

The BBI-SS PAS should not be cold started when the associated IMS control region is up. If a cold start of the BBI-SS PAS is done, an IMS control region outage could occur.

An automatic cold start occurs after:

- An IPL
- BBI maintenance that affects BBI data in CSA (any module name with a prefix of DM) is applied
The previous BBI-SS PAS was started with START=KILL

**Note**  
(MainView AutoOPERATOR only) The VPOOL parameter in SSJCL can be used to reset shared variables after a cold start except for an automatic cold start where a reset is forced after an IPL. The VPOOL parameter is used as follows:  
VPOOL=RESET | NORESET  
The default is NORESET.

If a warm start is attempted and the version of the BBI-SS PAS being started is not compatible with the version of the previous BBI-SS PAS used, the operator is prompted with a message that the releases are not compatible. The operator can then specify COLD or CANCEL.

**Note**  
START=FREE frees all allocated CSA in the same manner as a cold start. However, the BBI-SS PAS terminates immediately after the storage is freed.

## Starting a BBI-SS PAS before JES (MainView AutoOPERATOR only)

If you want MainView AutoOPERATOR to automate functions that are normally performed before JES is operational, the MainView AutoOPERATOR subsystem must be started before JES.

### To start the BBI-SS PAS before JES

1. Allocate a minimum of three dump data sets with these characteristics:  
   
   ```
   RECFM=VBA,LRECL=125,BLKSIZE=1632,DSORG=PS
   ```
   
   Make each one large enough to accommodate the largest dump that you expect.

2. Add the following allocations to your BBI-SS PAS JCL, pointing to the dump data sets that were created in the previous step.
   
   ```
   //BBDUMP01 DD DISP=SHR,DSN=dumpdsn1.VOL=SER=....,UNIT=....
   //BBDUMP02 DD DISP=SHR,DSN=dumpdsn2.VOL=SER=....,UNIT=....
   //BBDUMP03 DD DISP=SHR,DSN=dumpdsn3.VOL=SER=....,UNIT=....
   //SYSTSIN DD DUMMY
   //SYSTSPRT DD DUMMY
   ```

   **Note**  
The SYSTSIN and SYSTSPRT DD statements are required for IBM REXX support.
3 Catalog all data sets referenced in the SSJCL procedure in the master catalog or change all allocations to UNIT/VOLSER allocations.

4 Store the procedure in the data set SYS1.PROCLIB. Do not store it in a user proclib supported by JES.

5 Add a TIME parameter to the EXEC statement in BBSAMP member SSJCL.

6 Specify the type of JES that will be started after the BBI-SS PAS is active.

   To specify the JES type, use the JESTYPE= keyword in the BBISP*xx parmlib member (for more information, see the BBISP*xx parameters for MainView AutoOPERATOR topic in the MainView Customization Reference).

   The default is JES2. If the default is used on a JES3 system, event types EOS, EOM, and TLM will not function correctly and will not trigger any rules of these types.

7 Start the BBI-SS PAS with the following command:

   S SSJCL.id,SUB=MSTR

   The BBI-SS PAS opens the first dump data set when the BBI-SS PAS starts. This data set closes and the next one opens after an abend occurs. When the last one is opened and used, it is reopened and reused over and over until dumps are reset by using the BBI control command .RESET DUMP. You can issue this command from a terminal session, the system console, or an EXEC in the standard BBI command format.

   **Note**

   The terminal session can be started only after JES has initialized. If the MainView AutoOPERATOR BBI-SS PAS is started prior to JES, the IMFEXEC SUBMIT command cannot be used.

---

**Using the STALL interval parameter**

BBPARM member BBISP nn contains the following parameter for specifying the elapsed-time stall interval for overall BBI initialization:

   STALL=nnn

   where nnn is the number of seconds (elapsed time) before the BBI-SS PAS recognizes a stall condition. The default is 180 seconds. When the stall limit is reached, the BBI-SS PAS terminates with the following message:

   SS0924E STALL DETECTED DURING INITIALIZATION
Many BBI-SS PAS-to-BBI-SS PAS communication links can require significant time to complete. Use the STALL parameter to raise the time limit.

**Invoking timer-driven services at BBI-SS PAS startup**

At BBI-SS PAS startup, a group of user-defined timer-driven requests can be invoked automatically if the following conditions are met:

- The requests are defined in a member of BBPARM.
  The BBPARM member can contain predefined requests to start services or logging requests.
- The member name is specified with the BLK parameter in BBPARM member BBIISP00.

A separate request can be specified for each target.

**Displaying the status of a BBI-SS PAS**

You can check the status of a BBI-SS PAS and its resources by using the BBI control command DISPLAY ACTIVE.

Figure 5 on page 62 is an example of the result of a DISPLAY ACTIVE command for a BBI-SS PAS.

**Figure 5: Example status of a BBI-SS PAS**

```
BBI-SS STATUS AT hh:mm:ss ON SYSA  where SYSA is the SMFID
BBI VERSION 260 STARTED ON dd-mmm-yy AT hh:mm:ss
MONITORS  DEFINED - 030, ACTIVE - 010
LINKS     DEFINED - 003, ACTIVE - 002
USERS     DEFINED - 020, ACTIVE - 005
IMAGE LOG - ACTIVE - ON DATASET 1
JOURNAL   - ACTIVE - ON DATASET 1
```

See “BBI control commands” on page 77 for more information about the DISPLAY command.

**Stopping a BBI-SS PAS**

The BBI-SS PAS is terminated under two conditions:

- System shutdown
- BMC product or target system configuration modification, which might require changes to certain BMC parameters

A BBI-SS PAS can be either a started task or a job. If the BBI-SS PAS is a started task, you can terminate the BBI-SS PAS with a STOP command against the started task ID (stepname). If the BBI-SS PAS is a job, you can terminate the BBI-SS PAS with a CANCEL or STOP command; an example follows:

`P procname`

where `procname` is the job name or started procedure name of the BBI-SS PAS.

Or, if the PAS was started with an ID, issue

`P id`

**Note**

Normal or abnormal termination of a target subsystem does not affect BBI-SS PAS operation. When a target subsystem terminates, active monitors or timer-driven service requests for that target subsystem quiesce automatically. They restart when the target subsystem resumes operation. Any attempt to execute services that require access to an inactive target subsystem result in a message being issued.

---

**Controlling BBI-SS PAS resources**

This section describes the logging and control commands that are shared by the products in the BBI-SS PAS.

**Tip**

The information in this section does not apply to MainView VistaPoint (for CICS, DB2, DBCTL, and IMS workloads).

---

**Logging**

BBI records system commands, responses, and messages in a BBI-SS PAS journal log data set.

It records a user's terminal session commands and responses in a journal log data set. Two data sets are allocated for the BBI-SS PAS and terminal session journals. MainView for CICS, MainView for IMS Online, MainView for DBCTL, and MainView for DB2 use two image logs to record screen images produced by timer-driven services.
This section describes:

- What is recorded in the logs and when
- How to view, display, switch, and print a log
- Log maintenance and recovery

The record formats for the journal and image logs are described in “Journal and image log record formats” on page 297.

**Note**

BBI-SS PAS journal records can also be written to MainView Logger. For more information about

- Implementing this optional logging, see the discussion of BBPARM member BBISSPx in the MainView Customization Reference
- Displaying journal records, see “Displaying logged messages” on page 140

### BBI-SS PAS journal log

A BBI-SS PAS journal log consists of two log data sets, BBJRNL1 and BBJRNL2, which are used alternately.

BBI-SS PAS journal logging begins automatically when the BBI-SS PAS is started, unless the corresponding DD statements are removed from the BBI-SS PAS JCL.

A BBI-SS PAS journal log records:

- All BBI commands and responses issued from terminal session users assigned to the BBI-SS PAS
- All commands and responses issued automatically by MainView AutoOPERATOR EXECs
- Time stamps for BBI-SS PAS and target system start and stop
- Any warning or exception warning message issued by services
- BBI informational, error, and audit messages
- Service commands and messages
- DB2 commands and messages (MainView for DB2 only)
When messages are written to the system console, such as WTOs (write-to-operators), the BBI-SS PAS subsystem ID is appended whenever possible. The subsystem ID is not recorded in the BBI-SS PAS journal log.

**Viewing a BBI-SS PAS journal log online**

Use the LOG DISPLAY option from the Primary Option Menu to view a BBI-SS PAS journal log online. The log displayed is identified by the entry in the **TGT** field, which you can change. You can scroll the display left and right.

- If a target system ID is entered, the journal log of the BBI-SS PAS assigned to the target is displayed.

- If a BBI-SS PAS ID is entered, the journal log of that BBI-SS PAS is displayed.

The TGT value can be modified. When the value is changed, the BBI-SS PAS journal belonging to that specified target is displayed.

**Enhanced journal facility**

The Standard Journal Facility displays all messages from the BBI-SS PAS assigned to the specified target (default). However, the Enhanced Journal Facility provides user-customizable views of the Log Display. It allows you to view subsets of the journal log defined by one or more message origins.

The PROFILE command displays an input panel where you can define which messages you want to see. You can specify up to six patterns for message origins to be included and six patterns for message origins to be excluded in the display. Generic qualifiers can be used to define these patterns. For example, you can include all messages from CICS* and exclude all messages from CICSTEST.

You also can include messages from the current target specified in the **TGT** field by specifying the variable, &target, in the Included Origins column.

These specifications are saved in userid.BBPROF member LDPARM00, which enables each user to have an individual application profile.

To view the origin of the messages in the Log Display, scroll to the left.

---

**Note**

- You must specify JOURNAL=ENHANCED in BBPARM member BBISSP00 for the target BBI-SS PAS for the PROFILE specifications to take effect.

- The PROFILE specifications are not active for a target of LOCAL.
Displaying the status of a BBI-SS PAS journal log

To display the status of a BBI-SS PAS journal log, use the BBI control command DISPLAY JOURNAL (see “BBI control commands” on page 77 for more information about the DISPLAY command).

Switching BBI-SS PAS journal log data sets

When one BBI-SS PAS journal log data set becomes full, logging is switched automatically to the alternate data set.

Logging also can be switched manually to the alternate data set with the BBI control command SWITCH JOURNAL. This command could be used, for example, to synchronize the data sets with a target session (see “BBI control commands” on page 77 for more information about the SWITCH command).

Whenever a journal log data set is switched, either automatically or manually, a user exit (BBIUSR01) is invoked automatically to dump and reset the data set so that it is available for reuse (see the MainView Customization Reference for more information).

Note
If both journals are full, the switch process is not completed and logging stops.

Printing a BBI-SS PAS journal log

You can print a BBI-SS PAS journal log in the following ways:

- Automatically
  User exit BBIUSR01 is scheduled automatically to invoke a started task that prints and resets the BBI-SS PAS journal log data set (see the MainView Customization Reference).

- Manually by copying BBSAMP member DLOGJCL
  Edit the copied sample procedure to replace the PROC statement with a JOB statement. Submit the job to print and reset the BBI-SS PAS journal log data set so it can be reused.

Note
To print the BBI-SS PAS journal log to a data set on disk or tape, the BBIPRINT DD statement can be modified to point to a data set with DCB=(RECFM=FBA,LRECL=121) specified. Any suitable BLKSIZE is allowed.
Manually by copying BBSAMP member PLOGJCL

Edit the copied sample JCL and submit the job to print and *not* reset the BBI-SS PAS journal log data set. The data set must first be closed.

**Restricting BBI-SS PAS journal log messages and commands**

Use the journal log user exit routine SMLXIT to selectively restrict the messages and commands that go to the BBI-SS PAS journal log. The message text, however, cannot be altered.

The exit must be link-edited as reusable with the name SMLXIT. The following conventions are used:

- **Upon entry, register 0** points to one of the following values:
  - the VTAM node name that issued the command or received the message
  - the job name of the target system that sent a message to the BBI-SS PAS
  - the user ID that issued the command
  - the subsystem name that responded to the command

- **Register 1** points to one of the following locations:
  - +0 points to the length (two bytes) of the command or message text
  - +4 points to the actual text to be edited

- **Upon exit, all registers are restored except for register 15, which contains one of the following return codes:**
  - **RC=0** Log the message or command to the BBI-SS PAS journal log.
  - **RC=4** Do not log the message.

---

**Note**

Do not call other routines from this exit. The message cannot be changed.

A coding example of the journal log exit is supplied in BBSAMP member SMLXIT. Make any required changes, and then assemble and link-edit SMLXIT into the load library. If SMLXIT is in the load library, the exit is loaded at BBI-SS PAS initialization time. The following message is displayed on the console and written to the BBI-SS PAS journal log:

`SM0340I USER JOURNAL LOGGING EXIT LOADED`
Handling BBI-SS PAS journal log maintenance and errors

When the journal log is full or an I/O error is encountered on the journal, a user exit named BBIUSR01 is invoked (see the MainView Customization Reference).

When an error occurs, a message is sent to the BBI-SS PAS journal log for user action. Logging can be resumed by issuing the BBI control command START JOURNAL described in “BBI control commands” on page 77. When a BBI-SS PAS journal log is restarted after being stopped for any reason, recording resumes where it left off when the journal was stopped.

Terminal session journal log and screen-print data set

The terminal session journal log is a local journal (BBIJRNLL) allocated in each UAS that can be used to record all user commands and system responses.

Terminal session screen images can be captured by using the PRINT (PF4/16) key and are recorded in a BBISPRNT data set. BBIJRNLL and BBISPRNT are allocated with the MainView CLIST, as described below.

Allocating the terminal session journal or screen-print data set

1. A local journal log for user commands and system responses (BBIJRNLL) and a screen print log (BBISPRNT) can be allocated to a UAS.

   The terminal session journal (BBIJRNLL) and screen print data set (BBISPRNT) are allocated by the following MainView CLIST parameters:

   JOURNAL (YES)

   PRINT(YES)

   **Note**

   Logging occurs only if you specify LOGGING=YES in BBITSP00 and JOURNAL=YES in the MainView CLIST.

The contents of the BBISPRNT data set can be printed later (see the sample JCL in BBSAMP member SLOGJCL).

The terminal session journal log wraps around when full. The log reinitializes and recording starts at the beginning of the data set. All information previously recorded on the data set is lost.

When a journal log restarts after the terminal session stops for any reason, log recording resumes where it was before the terminal session stopped.
Viewing a terminal session journal log online

1 Use the Log Display option from the Primary Option Menu to view a terminal session journal log online. The log displayed is identified by the entry in the TGT field, which can be changed. Enter LOCAL in the TGT field to display the terminal session journal log.

Printing a terminal session journal log

1 Copy the sample procedure in BBSAMP member DLOGJCL.

2 Edit the copy that you created:
   a Replace the PROC statement with a JOB statement.
   b Change the BBIJRN1 data set names to reference the terminal session journal log data set.

3 Submit the job.

BBI-SS PAS image log

This topic applies only to the following products:

- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online

A BBI-SS PAS image log consists of two log data sets, BBIIMAG1 and BBIIMAG2, which are used alternately. BBI-SS PAS image logging begins automatically when the BBI-SS PAS is started unless the corresponding DD statement is removed from the BBI-SS PAS JCL. The MainView Customization Reference describes how these logs are allocated.

---

Note

These statements must be changed from comments to active DD statements.

---

BBI logic tests the space available in the image log data set before the image is added. If sufficient space is not available, BBI invokes the CLOSE and OPEN routines for the image log. As a result, even though a secondary extent may exist, it will not be used.
If logging stops for any reason, it can be resumed by issuing the BBI control command START IMAGE, as described in “BBI control commands” on page 77.

A BBI-SS PAS image log records screen images that are produced automatically by timer-driven analyzer and monitor services.

**Determining capacity of the BBI-SS PAS image log**

Images logged to the data set are of two types with different lengths. Thus, the capacity of the log depends on the combination of records that are logged to the data set. As a guideline, the image log uses 4092 bytes to store one large image—that is, 10 per 3380 track.

**Displaying the status of a BBI-SS PAS image log**

The status of a BBI-SS PAS image log can be checked with the BBI control command DISPLAY IMAGE (see “BBI control commands” on page 77 for more information about the DISPLAY command).

**Switching BBI-SS PAS image log data sets**

When one BBI-SS PAS image log data set becomes full, logging automatically switches to the alternate data set.

Logging can also be switched to the alternate data set manually with the BBI control command SWITCH IMAGE. This command could be used, for example, to synchronize the data sets with a target session, (see “BBI control commands” on page 77 for more information about the SWITCH command).

Whenever logging switches, either automatically or manually, a started task BBIUSR02, is invoked to print the data set (see the *MainView Customization Reference*). Any data residing on the alternate data set is overwritten.

**Printing a BBI-SS PAS image log**

You can print a BBI-SS PAS image log in the following ways:

- Automatically
  The started task, BBIILOG, is a procedure that can be scheduled automatically to print a full BBI-SS PAS image log data set.

- Manually
  The sample procedure in BBSAMP member ILOGJCL can be edited to replace the PROC statement with a JOB statement and submitted to print a BBI-SS PAS image log data set. Selection statements can also be defined, as described in “Selecting and printing image log records” on page 72.
Handling BBI-SS PAS image log maintenance and errors

When the image log data set is full or an I/O error is encountered on the image log, a user exit named BBIUSR02 is invoked (see the MainView Customization Reference).

After the error is corrected, logging can be resumed by issuing the BBI control command START IMAGE (see “BBI control commands” on page 77 for more information about the START command).

When a BBI-SS PAS image log is restarted after being stopped for any reason, it resumes recording at the beginning of the log data set identified as BBIIMAG1. Any data on BBIIMAG1 is overwritten.

Terminal session image log

A terminal session image log consists of one log data set, BBIIMAGL (ddname BBIIMAG1).

Tip

The information described in this topic applies to the following products only:

- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online

Terminal session image logging begins automatically when the terminal session is started unless one of the following conditions is met:

- Symbolic parameter IMAGE(NO) is used in the MainView CLIST
- Corresponding DD statement is removed from the MainView CLIST
- Parameter IMAGE=NO is specified in BBPROF member BBITSP00

A terminal session image log records BBI screen images produced by the terminal session.

The terminal session image log wraps around when full. The log is reinitialized and recording is started at the beginning of the data set. All information previously recorded on the data set is lost.
If the BBI-SS PAS is restarted while terminal session image logging is active, terminal session image logging will also restart at that time. All information previously recorded in the data set is lost.

**Printing a terminal session image log**

A terminal session image log can be printed by copying the sample JCL in BBSAMP member ILOGJCL, editing it to replace the PROC statement with a JOB statement, and submitting the job. Selection statements can also be defined, as described in “Selecting and printing image log records” on page 72.

**Selecting and printing image log records**

The program IMRPRINT may be executed from the BBLINK library to select and print the image records (X’FB’) from the BBI image logs (BBI-SS PAS or terminal session).

(Sample JCL is in BBSAMP member ILOGJCL.) If no selection is specified, all records are printed in chronological sequence.

The image log record contains a screen image that is produced by a display service. It might be produced as a record of a terminal request on the terminal session image log. If the screen image is produced as the result of a timer-driven service request, it is logged to the BBI-SS PAS image log. This second type of image might be either an informational display or a plot display of the historical data that was collected by a monitor service.

Each display record that is selected for printing is printed as it appears on the screen, except for the date and time fields (which are supplied on output to the terminal). The target name is shown on line 2 of the screen image. Additional information is printed to the right of the service display.

The last line, SELECTION SATISFIED: nn, appears only if selection was specified.

| DATE  yy/ddd | TIME  hh:mm:ss.s |
| SUBMITTING USERID: LNX001 |
| SELECTION SATISFIED: 01 |

An additional option, if selection is specified, allows the selected records to be written out to the file SELWRITE in addition to, or instead of, being printed. These records then can be sorted as needed and used again as input to IMRPRINT for printing in sequence. The satisfied SELECT statement number is put in the MSGSELNR field on output so that it is available for sorting.
At the completion of processing, the totals are printed as shown in the following example. If no selection is specified, only the first four lines appear.

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL LOG RECORDS READ:</td>
<td>1,528,799</td>
</tr>
<tr>
<td>TOTAL AUDIT RECORDS READ:</td>
<td>127</td>
</tr>
<tr>
<td>TOTAL AUDIT RECORDS PRINTED:</td>
<td>127</td>
</tr>
<tr>
<td>TOTAL AUDIT RECORDS WRITTEN:</td>
<td>54</td>
</tr>
<tr>
<td>NUMBER SELECTED, SELECT 01:</td>
<td>73</td>
</tr>
<tr>
<td>NUMBER SELECTED, SELECT 02:</td>
<td>54</td>
</tr>
</tbody>
</table>

Job control language for IMRPRINT is shown in Figure 6 on page 73.

**Figure 6: JCL for IMRPRINT**

```plaintext
//PRINT EXEC PGM=IMRPRINT[,PARM='UC']
//LOGFILE DD DSN=<BBI Image log>,DISP=OLD
//LIST DD SYSOUT=A
//SELECT DD *

select records (optional)
/*
//SELWRITE DD DSN=<AUDIT RECS>,DISP=NEW,(or DUMMY)
// DCB=(RECFM=VB,LRECL=2332,BLKSIZE=nnnn)
//
```

**Note**

PARM=UC is required only if the screens are to be printed in uppercase only.

**Selecting image records**

The 80-byte records are selected with a SELECT statement. If none are selected, all BBI audit records are printed.

The selection syntax is free format and keyword-oriented:

- Maximum of 80 columns can be used.
- Statements can be started anywhere and continued on any number of records, but symbols cannot be broken at position 80 and continued on the next statement.
- Keywords need not begin a new record.
- Any number of blanks, commas, slashes, or parentheses (which are ignored) can be interspersed in the text between symbols.
- Specify comments between asterisks (for example, a comment is begun by one asterisk and closed when another asterisk or the end of that record is found).
Note
If the input records are line numbered, place an asterisk after the text so the number is considered a comment.

A maximum of 20 SELECT statements can be entered:

- The logical condition OR applies between them.
- An audit record is printed only once (the first time it satisfies a SELECT statement). However, it can be sent to the SELWRITE file multiple times, once for each SELECT statement that it satisfies.
- The default is to PRINT the record.

The logical condition AND applies between selection types (for example, LTERM or TIME) within one SELECT statement:

- Any number of types can be specified in any order, but none can be duplicated within one SELECT (that is, specify all operands of one type together).
- No select types are required on a SELECT statement.

The logical condition OR applies between the operands or operand ranges specified for one type:

- The total number of operands in all SELECT statements must not exceed 500.
- Within this limit, any number of operands or operand ranges can be specified for one selection type.

**SELECT statement**

The syntax of the SELECT statement is as follows:

```
SELECT[PRINT|NOPRINT]keywords
```

or

```
SELECT[WRITE|NOWRITE]keywords
```

Table 8 on page 75 lists the valid SELECT keywords and operands.

Note
SERVICE=PLOT can be used to select all graphic display records. REQID, not SERVICE, is used to select specific monitor displays.
### Table 8: SELECT keywords and operands

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Description</th>
<th>Operands</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SERVICE</strong></td>
<td>Analyzer service name</td>
<td>Two- to five-character, alphanumeric service name and an optional one- to eight-character, alphanumeric parameter. The two fields must be separated by a hyphen (for example, SERVICE=DB2ST-RATE). If no parameter and no hyphen are specified, all images that match the service name are selected. If a hyphen is specified followed by a blank, only those images that match the service name and have no parameter are selected.</td>
</tr>
<tr>
<td><strong>REQID</strong></td>
<td>Monitor service request identifier</td>
<td>Two- to five-character, alphanumeric request ID and a one- to eight-character, alphanumeric parameter. The two fields must be separated by a hyphen (for example, REQID=SQLAC-CTL). The parameter must match the SET request parameter that is used to create the image.</td>
</tr>
<tr>
<td><strong>DATE</strong></td>
<td>Julian date</td>
<td>Numeric, five digits. The format is <em>yyddd</em>, where <em>yy</em> is the year and <em>ddd</em> is the Julian day (001-366).</td>
</tr>
<tr>
<td><strong>TIME</strong></td>
<td>Time of day</td>
<td>Numeric, four digits. The format is <em>hhmm</em>, where <em>hh</em> can be 00 to 24 and <em>mm</em> can be 00 to 59.</td>
</tr>
<tr>
<td><strong>USERID</strong></td>
<td>Submitting USERID</td>
<td>One- to eight-character, alphanumeric symbol</td>
</tr>
<tr>
<td><strong>USRQUAL</strong></td>
<td>Submitting USERID qualifier</td>
<td>One- to seven-character, alphanumeric symbol</td>
</tr>
</tbody>
</table>
Keywords | Description | Operands
--- | --- | ---
COMPLET | Service completion code mnemonic | Two-character, alphanumeric symbol
Valid entries are:
- OK (successful)
- NV (invalid request)
- SC (security violation)
- IF (IMS request failed)
- OF (OS request failed)
- AB (service abended)
- NP (invalid parameter)
- NG (service unsuccessful, not OK)

LTERM | Submitting user ID qualifier | One- to eight-character, alphanumeric symbol

Each input record is printed on the LIST file as it is read. When a new SELECT statement is recognized, a message with the assigned sequential select number is printed.

If any errors are found in the selection statements, error messages are written to the LIST file and the program terminates with a condition code of 20 (review the messages for this code). All statements are edited as much as possible.

**Examples**

- Select all LOCK, ULOCK, and PLOT requests entered from the terminal with ID AAB01 specified and print them. Also, select all plots logged asynchronously by timer requests submitted from the same terminal.
  ```sql
  SELECT SERVICE=LOCK,ULOCK,PLOT USERID=AAB01
  ```

- Select all requests for service DB2EX and write them to the SELWRITE file.
  ```sql
  SELECT WRITE NOPRINT SERVICE=DB2EX
  ```

- Select the requests that were made between 10:00 A.M. and 12:00 P.M. and submitted from the terminal with ID AAB01. Also, select those requests for the services DMON and DWARN from any terminal. Print is the default.
  ```sql
  SELECT TIME=1000-1200, USERID=AAB01
  SELECT TIME=1000-1200, SERVICE=DMON,DWARN
  ```
Select all requests to be printed. Write to the file any requests made from terminals with a name beginning with ABC, or any for a DB2ST, EDMPL, or BFRPL service.

```
SELECT(PRINT)SELECT WRITE USRQUAL=ABC
SELECT WRITE
SELECT WRITE SERVICE=(DB2ST,EDMPL,BFRPL)
```

**Note**

If a service that is entered from user ID ABC001 requests DB2ST, this audit record is written twice. Although the PRINT option was not turned off in SELECT statements 02-03, all requests are printed only once, selected by the first SELECT statement.

Select all PLOT displays of the timer-driven monitor request BPUTL-BP0 (buffer pool utilization for pool BP0).

```
SELECT REQID=BPUTL-BP0
```

**Selection edit messages**

The selection edit messages are as follows:

```
POS nn:  SELECT STATEMENT nn DEFINITION STARTED
POS nn:  WARNING - INVALID CHARACTER IGNORED
POS nn:  ERROR - DUPLICATE OPTION
POS nn:  ERROR - KEYWORD REQUIRED, TEXT IGNORED UNTIL FOUND
POS nn:  ERROR - DUPLICATE SELECT TYPE, SKIP TO NEXT KEYWORD
POS nn:  WARNING - EQUALS MISSING AFTER TYPE
POS nn:  ERROR - NO OPERANDS FOUND FOR THIS TYPE
POS nn:  ERROR - RANGE SPECIFICATION INVALID
POS nn:  ERROR - OPERAND INVALID FOR THIS TYPE
POS nn:  ERROR - TOO MANY OPERANDS
SELECT RECORD(S) INVALID - PROCESSING TERMINATED
```

**BBI control commands**

BBI control commands are used to control resources and functions.
Tip
The information described in this chapter applies to the following products only:

- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for MQ

These commands log the status of and modify the BBI-SS PAS online system. As described in “Issuing a BBI control command” on page 94, BBI control commands can be issued from a terminal session, an operator console, or a MainView AutoOPERATOR EXEC. For example, to log the status of the target BBI-SS PAS to the journal log, the DISPLAY ACTIVE command could be entered from a terminal session as follows:

```
.D A
```

The status information can then be viewed by using the BBI general service LOG DISPLAY.

To issue BBI control commands:

- Use a blank between commands and keywords.
- Use an equal sign, blank, or comma between keywords and parameters.

Table 9 on page 79 lists all the BBI control commands and uses the following notation conventions:

- Some commands, keywords, and parameters are shown in mixed case (such as Cancel). The uppercase letters (C, for example) represent a valid abbreviation.
- The phrase "not applicable" in the Parameters column means that there are no parameters for that command.
### Table 9: BBI control commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel</td>
<td>DeLaY</td>
<td>delay_ID</td>
<td>(MainView AutoOPERATOR only) Cancels all delayed actions (ALL) or a specific action specified by the Delay ID. Use the .DISPLAY DLY command to see a list of delayed actions. After you have identified a message from the Delay ID token, you can use the .CANCEL DLY command to cancel the delayed action.</td>
</tr>
<tr>
<td>Exec</td>
<td>eid</td>
<td>All</td>
<td>(MainView AutoOPERATOR only) Cancels the EXEC identified by eid (either running or queued) or cancels all EXECs. <strong>WARNING:</strong> Use extreme caution when using the ALL keyword as all EXECs that are running or queued at that point in time will be canceled.</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Display</td>
<td>Active</td>
<td>not applicable</td>
<td>Logs the status of the BBI-SS PAS</td>
</tr>
<tr>
<td>AOLOG</td>
<td>not applicable</td>
<td></td>
<td><em>(MainView AutoOPERATOR only)</em> displays the current status of the Automation Logger application</td>
</tr>
</tbody>
</table>
| DeLaY   | All | | *(MainView AutoOPERATOR only)* Displays a list of delayed actions sorted by the Delay ID token, which is assigned to a message when it is delayed. `commandType` is the type of delay you want to view. The following values are valid:  
  - WTO  
  - EXEC  
  - MQEXEC  
  - CMDCICS  
  - CMDSS  
  - CMDBBI  
  - CMDNETV  
  - CDMMQ  
  - CMDIMS  
  You can use the * wildcard with `commandType`. For example, CMD*, shows all delayed action types that start with CMD.  
  After you have identified a message from the Delay ID token, you can use the .CANCEL DLY command to cancel the delayed action. |
<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display (continued)</td>
<td>Exec</td>
<td>All</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gme</td>
<td>All</td>
<td>Conn [,node] Publish</td>
<td>Receive Trace [,nnnn]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Command

<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display  (continued)</td>
<td>Image</td>
<td>not applicable</td>
<td>(MainView for CICS, MainView for IMS Online, MainView for DBCTL, MainView for DB2) Logs the status of BBI-SS PAS image log</td>
</tr>
<tr>
<td></td>
<td>JESF</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Displays the current status of the JES filter setting (inactive or active)</td>
</tr>
<tr>
<td></td>
<td>Journal</td>
<td>not applicable</td>
<td>Logs the status of the BBI-SS PAS journal log and the MainView Logger</td>
</tr>
<tr>
<td></td>
<td>Keys</td>
<td>not applicable</td>
<td>Displays all the option password keys read from the BBKEYS member of BBPARM and their validation status</td>
</tr>
<tr>
<td></td>
<td>Link</td>
<td>All</td>
<td>ssid</td>
</tr>
<tr>
<td></td>
<td>MVLog</td>
<td>not applicable</td>
<td>Displays the current settings for MainView Logger and diagnostic information about its use</td>
</tr>
<tr>
<td></td>
<td>PARTNERS</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Displays the status of the TapeSHARE partners in the TapeSHARE PLEX</td>
</tr>
<tr>
<td></td>
<td>Products</td>
<td>prdname</td>
<td>ACTIVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>prdname is the name of the product.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ACTIVE indicates that the product was initialized properly and was not stopped.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>INACTIVE indicates that the product was initialized properly at BBI-SS PAS startup, but it was stopped or was not found (not initialized properly).</td>
</tr>
<tr>
<td></td>
<td>Remote</td>
<td>not applicable</td>
<td>Displays status of all remote terminal session users connected to this BBI-SS PAS through a remote BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>RULES</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Shows whether MainView AutoOPERATOR Rule processing is enabled or disabled</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------</td>
<td>------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Display (continued)</td>
<td>RULESET</td>
<td>[,type],</td>
<td><em>(MainView AutoOPERATOR only)</em> Displays information about Rule Sets and Rules</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[rulesetid],</td>
<td>■ <em>type</em> identifies the type of data to be shown; this value is optional and defaults to ENA if not specified</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[start],</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[count]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The following lists all of the possible valid values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ ENA: show enabled Rule Sets</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ DIS: show disabled Rule Sets</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ TST: show Rule Sets in TST mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ LOD: show Rule Sets in LOADED mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ ALL: show Rule Sets of all types</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The <em>rulesetid</em> must start with RUL or AAORUL to be valid; for example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ RUL*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ AAORUL*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ AAORULA1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ <em>start</em> the relative starting Rule Set number; this value is optional and defaults to 1 (start with the first matching Rule Set) if not specified</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ <em>count</em> the number of Rule Sets or Rules to display; valid values from 1 to 75; this value is optional and defaults to 20 if not specified</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Use this parameter to limit the number of lines in the display and prevent flooding the Journal or SYSLOG with returned values.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>You can omit (or skip) parameters by using a comma as a placeholder. The following list shows some sample commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ .D RULESET (displays the first 20 enabled Rule Sets)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ .D RULESET,DIS,AAORUL* (displays up to 20 disabled Rule Sets starting with the first one found that matches the characters AAORUL)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ .D RULESET,,15 (displays up to 20 enabled Rule Sets starting with the 15th one)</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------</td>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Display (continued)</td>
<td>SOF</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Displays the Shared Object Facility</td>
</tr>
<tr>
<td>SSR</td>
<td>not applicable</td>
<td>Displays subsystem resource usage, including CPU usage for each exit</td>
<td></td>
</tr>
<tr>
<td>User</td>
<td>ALL</td>
<td>name</td>
<td>Logs status of all terminal session users assigned to this BBI-SS PAS</td>
</tr>
<tr>
<td>Vpool</td>
<td>ALL</td>
<td>name</td>
<td><em>(MainView AutoOPERATOR only)</em> Displays the SHARED (default) or PROFILE variable pool statistics for the named variable or for all variables</td>
</tr>
<tr>
<td>XCF</td>
<td>not applicable</td>
<td>Displays the other BBI-SS PASs that this PAS is connected to, their cross-system coupling facility (XCF) member names, and how many requests are pending on this connection</td>
<td></td>
</tr>
<tr>
<td>Get</td>
<td>dddd</td>
<td>ONLINE GIVE</td>
<td><em>(MainView AutoOPERATOR only)</em> Allows you to VARY the status of a tape device online where dddd is the device address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NOGIVE</td>
<td>Valid parameters are</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TAKE</td>
<td>Valid parameters are</td>
</tr>
<tr>
<td>Help</td>
<td>ALL</td>
<td>command</td>
<td>Displays syntax information and examples for the BBI control commands</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not applicable</td>
<td>You can request information for a specified command, (.HD) or for all commands (.H ALL). The default is ALL.</td>
</tr>
<tr>
<td>Locate</td>
<td>U,ddd</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Locates and displays the status of a device, where dddd is the device address</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>rEset</td>
<td>Auth</td>
<td>ALL</td>
<td>userid (BMC proprietary security only) Re-creates the BBPARM data set authorization of all terminal session users or one specific user on this BBI-SS PAS If ALL is specified, all authorization is re-created for local and remote users, as well as the generic authorization that is defined by $GENERIC, $USERID, and $RMTID.</td>
</tr>
<tr>
<td>Bldl</td>
<td>SYSPROC</td>
<td></td>
<td>(MainView AutoOPERATOR only) Reissues the BLDL for the data set (BBPROC) that is allocated by DD name SYSPROC to activate EXEC members that are added or to delete members The RESET command is not required when existing members are modified because the changes take place immediately.</td>
</tr>
<tr>
<td>BM</td>
<td>DB2_id</td>
<td></td>
<td>(MainView for DB2 only) Stops and restarts the connection between MainView for DB2 and the specified target DB2 subsystem</td>
</tr>
<tr>
<td>Dumps</td>
<td>not applicable</td>
<td></td>
<td>Reuses the dump data set until a reset dump command is issued from a terminal session Dump data sets are cleared only when specified in the SSJCL.</td>
</tr>
<tr>
<td>Exec</td>
<td>not applicable</td>
<td></td>
<td>(MainView AutoOPERATOR only) Resets the statistics for running and queued EXECs</td>
</tr>
<tr>
<td>FOCal</td>
<td>not applicable</td>
<td></td>
<td>(MainView FOCAL POINT only) Stops and restarts MainView FOCAL POINT without shutting down the BBI-SS PAS</td>
</tr>
</tbody>
</table>
### Command: rEset (continued)

**Keywords**: Gme  
**Parameters**: nodename | connection_name  
**Description**: `nodename` stops and restarts communication with a GME server node by using the parameters that are specified in the active AAOGMExx member.  
`nodename` can be one of the following values:

- **TGTNAME** is the name of the GME server node whose connection is to be stopped and started. Typically, this name will be the workstation ID of the system where the remote GME server node is running.
- **IP Address:Port** is a combination of the IP address and the port number connected by a colon (:).
- **Hostname:Port** is a combination of the host name and the port number connected by a colon (:).

`connection_name` stops and restarts communication with a BMC Impact Manager cell by using the parameters that are specified in the active AAOGMExx member and the MCELL.DIR data set.

### Command: MVLog

**Keywords**: not applicable  
**Parameters**:  
**Description**: Stops the connection to the active instance of MainView Logger (if any), and starts a new connection to an instance of MainView Logger by using the current values of the MVLOG and MVLOGSS parameters in member BBISSPxx. If MVLOG=NO is found in BBISSPxx, .E MVLOG has the same effect as .P MVLOG (that is, MainView Logger is not restarted).
<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rEset</td>
<td>Parm</td>
<td>AAOALSxx</td>
<td><em>(MainView AutoOPERATOR only)</em> Reinitializes the parameters specified in</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>member AAOALSxx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOALTxx</td>
<td><em>(MainView AutoOPERATOR-to-PATROL Enterprise Manager Interface only)</em> Drops</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>all connections to PATROL Enterprise Manager and terminates the VTAM session</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>After processing of AAOALTxx, where xx can be any suffix, the VTAM session</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and all remote connections are established.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOARPxx</td>
<td><em>(MainView AutoOPERATOR only)</em> Reinitializes the parameters specified in</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>member AAOARPxx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOEXPxx</td>
<td><em>(MainView AutoOPERATOR only)</em> Reinitializes the parameters specified in</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>member AAOEXPxx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOGMExx,</td>
<td>Reinitializes the parameters specified in member AAOGMExx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RECYCLE</td>
<td>Valid options are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOGMExx,</td>
<td>• RECYCLE stops the entire GME component and then restarts it. All current</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RESYNC</td>
<td>connections are restarted, and new connections are established based on</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>the AAOGMExx member and the MCELL.DIR data set.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• RESYNC stops any active GME connections that might exist (but have been</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>removed from the active AAOGMExx member) and starts any new connections</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>that might have been added to AAOGMExx since the recycle or startup of</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MainView AutoOPERATOR.</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------</td>
<td>----------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>rEset</td>
<td>Parm (continued)</td>
<td>AAORULxx</td>
<td>Reinitializes the AAORULxx Rule Set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOTRNxx</td>
<td>(MainView AutoOPERATOR only) Reinitializes the parameters that are specified in member AAOTRNxx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAOTSPxx</td>
<td>(MainView AutoOPERATOR only) Reinitializes the parameters that are specified in member AAOTSPxx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMRCTG</td>
<td>(MainView for CICS only) Resets the current CICS Transaction Gateway (CTG) monitoring parameters to the parameters that are specified in member CMRBEXnn</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMRPRBT</td>
<td>(MainView for CICS only) Reinitializes a table that sets default and user threshold values for messages displayed with the PROBLEM service or background problem services</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMRSECU</td>
<td>(MainView for CICS and MainView AutoOPERATOR for CICS option only) Reinitializes the CMRSECU module and creates a table of security definitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CMRSOPT region</td>
<td>(MainView for CICS only) Reinitializes a startup table of monitoring parameters for the CICS environment</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The format of the command is .E PARM CMRSOPT region</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>where region is the name of the target CICS region to have its CMRSOPT reset.</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>rEset</td>
<td>Parm</td>
<td>DMRBEX00 or IMFBEX00</td>
<td>(MainView for DB2, MainView for IMS Online, and MainView for DBCTL only) DMRBEX00 resets the MainView for DB2 message logging, background sampler exception, and application trace parameters. IMFBEX00 defines the MainView for IMS Online and MainView for DBCTL parameters for application trace defaults, detail trace buffer allocation, excessive threshold values, shared queues data collection rates, and monitor dump values.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RULxxxxx reinitializes the RULxxxxx Rule Set</td>
</tr>
<tr>
<td>QManager</td>
<td>MQ_ssid</td>
<td></td>
<td>(MainView AutoOPERATOR only) Stops and restarts the connection to the specified WebSphere MQ queue manager. The WebSphere MQ queues that are being monitored by MainView AutoOPERATOR are closed and reopened.</td>
</tr>
<tr>
<td>RULes</td>
<td>not applicable</td>
<td></td>
<td>(MainView AutoOPERATOR only) Disables all currently active Rule Sets and re-enables the Rule Sets that are defined with the RULESET parameter in BBPARM member AAOPRMxx If you had manually enabled any Rule Sets after the BBI-SS PAS started, those Rule Sets are not restarted with this command.</td>
</tr>
<tr>
<td>Stats</td>
<td>not applicable</td>
<td></td>
<td>(MainView AutoOPERATOR only) Resets message and automation statistics</td>
</tr>
<tr>
<td>TS</td>
<td>VALIDATE</td>
<td></td>
<td>(MainView AutoOPERATOR only) Checks the path validity for all tape devices For example, suppose a path was marked as ACTIVE before this command is issued. If the path for a tape device becomes INACTIVE when the .E TS,VALIDATE command is issued, the entry in the device table for that tape device will be marked as INACTIVE.</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>seT</td>
<td>DAE</td>
<td>[ON</td>
<td>OFF]</td>
</tr>
</tbody>
</table>
|         | DUMPS=   | [YES | NO | SDUMP] | (**BBI-SS PAS only**) Controls the creation of BBI-formatted dumps  
(If SDUMP is specified, SVC dumps are created for the BBI-SS PAS.) |
| JESFLTR |          | [CMD | WTO]    | (**MainView AutoOPERATOR only**) Modifies the JESFLTR setting in member BBISSPxx |
| RULE    |          | [ENA | DIS | TEST] ruleID | (**MainView AutoOPERATOR only**) Enables, disables, or places a Rule into test mode by ruleID  
Examples of this command are:  
```
.T RULE,ENA,ICH00001  
.T RULE,TEST,ICH00012
``` |
| RULESET |          | [ENA | DIS | RES | SAV | TST], RuleSetName | (**MainView AutoOPERATOR only**) Enables or disables a specific Rule Set by RuleSetName  
You also can reset or save a Rule Set by RuleSetName.  
Resetting a Rule Set is the same as disabling and re-enabling a Rule Set.  
Specifying TST puts the specified Rule Set in TEST mode.  
Examples of this command are:  
```
.T RULESET,ENA,RUL01CSM  
.T RULESET,RES,AAORUL xx  
.T RULESET,SAV,AAORUL xx  
.T RULESET,TST,RULMQS xx
``` |
| RULESET= |          | RuleSetName | (**MainView AutoOPERATOR only**) Disables all the currently enabled Rule Sets and enables the Rule Sets specified by RuleSetName  
Prior to issuing the seT command, any Rules that you have created and enabled but have not saved to DASD will be lost when you issue the seT command.  
This command does not change the setting you specified in BBPARM member AAOPRMxx; it only lets you override the setting until the BBI-SS PAS is restarted. |
<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>AOLOG</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Attempts to start the connection to the Automation Logger PAS</td>
</tr>
<tr>
<td></td>
<td>BM</td>
<td>DB2_ID</td>
<td><em>(MainView for DB2 only)</em> Attempts to re-establish a connection between MainView for DB2 and the target DB2 subsystem ID. If the connection is successfully established, it restarts all background monitors for that subsystem.</td>
</tr>
<tr>
<td></td>
<td>Exec</td>
<td>(patternName)</td>
<td><em>(MainView AutoOPERATOR only)</em> Starts EXECs by using matching patternName (use * for all EXECs)</td>
</tr>
<tr>
<td></td>
<td>FOCal</td>
<td>not applicable</td>
<td><em>(MainView FOCAL POINT only)</em> Starts MainView FOCAL POINT, if it was previously stopped</td>
</tr>
<tr>
<td></td>
<td>Gme</td>
<td>nodename</td>
<td>nodename starts communication with a General Messages Exchange (GME) server node by using connection parameters that are read from the active AAOGMEExx member. For a list of possible values for nodename, refer to the description of RESET GME nodename in this table.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>connection_name</td>
<td>connection_name starts communication with a BMC Impact Manager cell by using the parameters that are specified in the active AAOGMEExx member and the MCELL.DIR data set.</td>
</tr>
<tr>
<td></td>
<td>GTS</td>
<td>[nn]</td>
<td><em>(MainView AutoOPERATOR and MainView for MQ only)</em> Starts the Generic TCP/IP Server (GTS) by using the parameters in BBPARM member BBTTCPnn, where nn is the suffix specified for the GTS=nn parameter in BBPARM member BBISSP00. If the GTS parameter is not specified in BBISSP00, member BBTTCP00 is used to start the GTS. To override the GTS parameter in BBISSP00, specify the suffix of an existing BBTTCPnn member (for example, $GTS,02$).</td>
</tr>
<tr>
<td></td>
<td>Image</td>
<td>not applicable</td>
<td><em>(MainView for CICS, MainView for IMS Online, MainView for DBCTL, MainView for DB2)</em> Starts image logging for the current BBI-SS PAS. Startup logic is the same as when a BBI-SS PAS is initially started.</td>
</tr>
<tr>
<td></td>
<td>JESconct</td>
<td>not applicable</td>
<td>Starts the JES reconnection process, as specified in member BBISSPxx (JESCNCT=YES)</td>
</tr>
<tr>
<td></td>
<td>Journal</td>
<td>not applicable</td>
<td>Starts journal logging for the current BBI-SS PAS. Startup logic is the same as when a BBI-SS PAS is initially started.</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Start (continued)</td>
<td>Link</td>
<td>ALL</td>
<td>ALL</td>
</tr>
<tr>
<td>MVLog</td>
<td>not applicable</td>
<td>not applicable</td>
<td>Starts a connection to an instance of MainView Logger and begins logging BBI-SS PAS journal records by using the current values of the MVLOG and MVLOGSS parameters in member BBISSP xx If MVLOG=NO is found in BBISSP xx, a message is issued that indicates MainView Logger logging is not available.</td>
</tr>
<tr>
<td>Rules</td>
<td>not applicable</td>
<td>not applicable</td>
<td>(MainView AutoOPERATOR only) Starts Rules processing</td>
</tr>
<tr>
<td>Command</td>
<td>Keywords</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>stoP</td>
<td>BM</td>
<td>$DB2_ID$</td>
<td><em>(MainView for DB2 only)</em> Terminates a connection between MainView for DB2 and the target DB2 subsystem ID</td>
</tr>
<tr>
<td></td>
<td>Exec</td>
<td>(patternName)</td>
<td><em>(MainView AutoOPERATOR only)</em> Stops the EXECs by using a specific patternName. For example, to disable scheduling of all EXECs that begin with the letter A, specify <code>.STOP EXEC A*</code> Use * for all EXECs. Issuing the command .RESET BLDL SYSPROC re-enables the scheduling of all EXECs.</td>
</tr>
<tr>
<td></td>
<td>FOCal</td>
<td>not applicable</td>
<td><em>(MainView FOCAL POINT only)</em> Stops MainView FOCAL POINT without shutting down the BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>Gme</td>
<td>nodename</td>
<td>nodename stops communication with a General Messages Exchange (GME) node. For a list of possible values for nodename, refer to the description of RESET GME nodename in this table. connection_name stops communication with a BMC Impact Manager cell.</td>
</tr>
<tr>
<td></td>
<td>GTS</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR and MainView for MQ only)</em> Stops the Generic TCP/IP Server (GTS)</td>
</tr>
<tr>
<td></td>
<td>Image</td>
<td>not applicable</td>
<td><em>(MainView for CICS, MainView for IMS Online, MainView for DBCTL, MainView for DB2)</em> Stops image logging for the current BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>JESconct</td>
<td>not applicable</td>
<td>Stops the JES reconnection process, as specified in member BBISSPxx (JESCNCT=NO)</td>
</tr>
<tr>
<td></td>
<td>Journal</td>
<td>not applicable</td>
<td>Stops journal logging for the current BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>Link</td>
<td>ALL</td>
<td>Stops the communication link between the BBI-SS PAS subsystem and the current BBI-SS PAS or all available BBI-SS PASs and the current BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>MVLog</td>
<td>not applicable</td>
<td>Stops the connection to the active instance of MainView Logger (if any)</td>
</tr>
<tr>
<td></td>
<td>Rules</td>
<td>not applicable</td>
<td><em>(MainView AutoOPERATOR only)</em> Stops Rules processing. The effect is similar to inactivating all active Rule Set entries.</td>
</tr>
<tr>
<td></td>
<td>swItch</td>
<td>Image</td>
<td><em>(MainView for CICS, MainView for IMS Online, MainView for DBCTL, MainView for DB2)</em> Switches from the current image log data set to the alternate and invokes a started task</td>
</tr>
<tr>
<td></td>
<td>Journal</td>
<td>not applicable</td>
<td>Switches journal logging to the alternate data set and invokes a started task. A switch cannot be made if the alternate data set is full.</td>
</tr>
</tbody>
</table>
Issuing a BBI control command

BBI control commands can be issued from:

- Terminal session
- Operator console
- MainView AutoOPERATOR EXEC

Where a BBI control command is issued determines which BBI-SS PAS the command operates against, and how the command is entered:

Terminal session

If the command is issued from a terminal session, it operates against the BBI-SS PAS that is identified in the TGT field, or against the BBI-SS PAS for the target that is identified in the TGT field. The target BBI-SS PAS can be either local or remote.

From a terminal session, the BBI control command must be prefixed with a period (.) to prevent conflicts with local application commands.

Operator console

If the command is issued from an operator console, it operates against the BBI-SS PAS that is assigned to the local target environment.

From an operator console, use one of the following:

- MODIFY (F) command
- subsystem command recognition character that is specified in BBPARM member BBISSP00
- MVS command prefix that is specified using the CMDPREF parameter in BBISSP00

Note
For more information about MainView AutoOPERATOR EXECs, see the MainView AutoOPERATOR Advanced Automation Guide.
To issue a BBI control command

- From a terminal session, use the **COMMAND** line of any full-screen application display panel and prefix the command with a period, as shown in the following example:
  `.LINK,SSB2`

- From an operator console, use the MODIFY (F) command or the subsystem command recognition character.

  - An example of a BBI control command with the MODIFY (F) command is as follows:
    ```
    F SSA1,D ACTIVE
    ```
    where
    - SSA1 is the BBI-SS PAS job name or started task ID.
    - D is the BBI control command (DISPLAY).
    - ACTIVE is a parameter of the DISPLAY command.

  - An example of the same BBI control command with the subsystem command recognition character is as follows:
    ```
    :D ACTIVE
    ```
    The subsystem command recognition character is defined with the CMDCHAR parameter in BBPARM member BBISSP00. If no character is specified (the default), the subsystem command recognition character cannot be used to issue BBI control commands.

    The command is processed by each BBI-SS PAS that has a matching subsystem command recognition character specified in its associated BBISSP00 member.

    **Note**

    Be sure that the subsystem command recognition character that you specify in BBISSP00 does not conflict with subsystem command recognition characters that are defined for other subsystems, such as JES.

    - Examples of the same BBI control command using the MVS command prefix (for example CMDPREF=ABCD?) are as follows:
      ```
      ABCD?D ACTIVE
      ABCD? D ACTIVE
      ```
      The example MVS command prefix ABCD? is defined using the parameter CMDPREF in BBISSP00. This prefix must be unique within the MVS LPAR and must not be a subset or superset of any pre-existing MVS command prefixes or subsystem command characters.
Logging BBI control commands and responses

All BBI control commands and responses are recorded in the BBI-SS PAS journal log. If the command is issued from an operator console, command responses are also returned to the originating console as single line or multiline WTOs.

BBI control command responses

The following responses are general to all BBI control commands:

- Short message response to terminal session:
  
  COMMAND ISSUED

- BBI-SS PAS journal log and console response:

<table>
<thead>
<tr>
<th>Code</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM9100I</td>
<td>BBI COMMAND ACCEPTED</td>
</tr>
<tr>
<td>IM9135E</td>
<td>BBI COMMAND REJECTED</td>
</tr>
<tr>
<td>IM9131E</td>
<td>INVALID BBI COMMAND Follows:  command string</td>
</tr>
<tr>
<td>IM9132E</td>
<td>COMMAND  command string</td>
</tr>
<tr>
<td>IM9133E</td>
<td>INVALID SYNTAX FOR COMMAND  command</td>
</tr>
<tr>
<td>IM9134E</td>
<td>INVALID PARAMETER FOR  resource * parameter *</td>
</tr>
<tr>
<td>IM9136E</td>
<td>ERROR EXECUTING COMMAND:</td>
</tr>
<tr>
<td>IM9132E</td>
<td>COMMAND  command string</td>
</tr>
</tbody>
</table>

To display the responses to commands that you entered, use the LOG DISPLAY option on the Primary Option Menu to view the BBI-SS PAS journal log.

To display a detailed explanation of any error message or abend code, use the MESSAGES option from the Primary Option Menu.
Managing cross-system communication

One of the principal benefits of MainView Infrastructure is the ability to monitor the performance of multiple MainView products from a single terminal session. You can see the current or historical performance of systems running simultaneously on all system images.

You can skip this chapter if:

- All CASs are running in the same sysplex and the default XCF group name of BBGROUP is acceptable.
  CASs communicate through the XCF link as long as they all have the same XCF group name, as described in the MainView Customization Reference.

- The current CAS is not communicating with another CAS on a different sysplex through a VTAM, TCP/IP, or XCF link.

Figure 7 on page 98 illustrates the following information:

- How MainView products communicate with each other between system images.
VTAM links that enable MainView products in a BBI-SS PAS on one system image to communicate with products on another system image.

**Figure 7: System communication links**

Managing CAS-to-CAS communication

CAS-to-CAS communication allows your terminal session to access information from PASs connected to another CAS, on another system image, or in a remote data center.

**Tip**
The information in this topic applies to all of the MainView products except MainView FOCAL POINT.

**Figure 7 on page 98** shows how MainView products communicate with each other between system images. Typically, a single CAS serves as a node that transfers information collected from all PASs running on a system image. XCF communication is established automatically between CASs running in the same sysplex and in the same XCF group. If you do not want two sets of CASs within a sysplex to communicate through XCF, you can change the default XCF group to a different value by using:

- **DFLTGRP=** parameter on the CAS startup JCL
- **CASDEF** administrative view
For example, if you have a production set and a test set, the test CASs can use a
default group name of TESTCAS while the production CASs use the default group
name BBGROUP.

VTAM communication is established between CASs that are running on separate
images outside of a sysplex environment. Optionally, you can establish TCP/IP
communication by specifying a port number and IP address on the CAS definition.

Table 10 on page 99 lists common tasks and the associated commands for
controlling cross-system communication links between a local CAS and a remote
CAS. Each task includes a page reference to more information about completing the
task.

Table 10: Cross-system communication management tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>View or Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage CAS definitions</td>
<td>See respective topics for commands</td>
<td>“Managing CAS definitions” on page 99</td>
</tr>
<tr>
<td>View the status of all CAS-to-CAS links</td>
<td>CASACT view</td>
<td>“Displaying active systems and sessions” on page 111</td>
</tr>
<tr>
<td>View information about the CAS you are running on and related components</td>
<td>CASINFO view</td>
<td>“Viewing information about your current CAS” on page 112</td>
</tr>
<tr>
<td>Start VTAM communication links</td>
<td>STart primary or ST line command</td>
<td>“Starting VTAM communication” on page 113</td>
</tr>
<tr>
<td>Quiesce VTAM communication links</td>
<td>QUIEsce primary or QUI line command</td>
<td>“Quiescing VTAM communication” on page 114</td>
</tr>
<tr>
<td>Halt VTAM communication links</td>
<td>HALt primary or HAL line command</td>
<td>“Halting VTAM communication” on page 115</td>
</tr>
</tbody>
</table>

Managing CAS definitions

To manage CAS definitions for cross-system communication, see the tasks listed in
the following table.

Table 11: CAS definition management tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>View or command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add CAS definitions for the first time</td>
<td>Not applicable</td>
<td>&quot;Defining CAS-to-CAS Communication&quot; in the MainView Customization Reference</td>
</tr>
<tr>
<td>Task</td>
<td>View or command</td>
<td>Reference</td>
</tr>
<tr>
<td>------</td>
<td>----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>View all CAS definitions</td>
<td>CASDEF view</td>
<td>“Viewing all CAS definitions (CASDEF view)” on page 100</td>
</tr>
<tr>
<td>View the details of one CAS definition</td>
<td>CASDEFD view</td>
<td>“Viewing CAS definition details (CASDEFD view)” on page 101</td>
</tr>
<tr>
<td>Obtain the edit lock to perform any command except INStall</td>
<td>EDIT primary command</td>
<td>“Editing CAS definitions” on page 102</td>
</tr>
<tr>
<td>Add a CAS definition after first-time customization</td>
<td>ADD primary command or A line command</td>
<td>“Adding CAS definitions” on page 102</td>
</tr>
<tr>
<td>Change an existing CAS definition</td>
<td>CHAnge primary command or C line command</td>
<td>“Changing existing CAS definitions” on page 104</td>
</tr>
</tbody>
</table>
| Define TCP/IP communication for a CAS | ADD primary command or A line command  
CHAnge primary command or C line command | “Defining TCP/IP communication” on page 105 |
| Cancel edit mode, adds, or changes in progress | CANcel primary command | “Canceling CAS definition edits” on page 106 |
| Delete a CAS definition | DELete primary command or DEL line command | “Deleting a CAS definition” on page 107 |
| Recover a CAS definition that was previously deleted | UNDelete primary command or UND line command | “Recovering a CAS definition” on page 107 |
| Immediately update the CAS with new or changed CAS definitions | INStall primary command or INS line command | “Immediately installing changes to CAS definitions” on page 108 |
| Permanently save changes to the UBBPARM parameter library CAS definition member (BBMTYB00) | SAVE primary command (or the END primary command) | “Saving CAS definitions” on page 109 |
| Deploy (copy) CAS definitions to other system | DEPloy primary command | “Deploying CAS definitions” on page 109 |
| Manage and monitor deployment | DPLYMNT view | “Managing and monitoring the deployment process” on page 110 |

**Viewing all CAS definitions (CASDEF view)**

To access the CASDEF view, perform one of the following tasks:

- In Plex Manager, type **CASDEF** on the **COMMAND** line and press **Enter**.

- From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the CASDEF view:

  ```
  CON * PLEXMGR;CASDEF
  ```
The CASDEF view is displayed, as shown in Figure 8 on page 101.

Figure 8: CASDEF view (left)

ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) --------------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
>W1 =CASDEF=SYSB==*========(00 BROWSE )====PLEXMGR==D====2
CMD CAS      Cur Description                           Status           VTAM
--- Name---- Sys -----------                           ---------------- ApplNam
C001     Yes Production CAS (VTAM)                 Installed        *NONE*
C002     No  TCP/IP Test CAS                       Installed        *NONE*

Note
The word Yes in the Cur Sys field identifies the local CAS.

You can scroll to the right to display the remaining fields of the CASDEF view, as shown in Figure 9 on page 101.

Figure 9: CASDEF view (right)

ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) --------------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
>W1 =CASDEF=SYSB==*========(00 BROWSE )====PLEXMGR==D====2
CMD CAS      VTAM     Port  Hostname        TCPIP    Del IP
--- Name---- ApplName Num-- ----------      Jobname- --- Address---
C001     *NONE*                                  N
C002     *NONE*    4104 172.017.008.092          N   172.017.008.092

For more information about any field in the CASDEF view, place the cursor on the field and press the Help key.

Viewing CAS definition details (CASDEFD view)

1 Display the CASDEF view, as described in “Viewing all CAS definitions (CASDEF view)” on page 100.

2 Place the cursor on the CAS definition that you want to see details about and press Enter.

The CASDEFD view is displayed, as shown in Figure 10 on page 101.

Figure 10: CASDEFD view

ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) --------------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
>W1 =CASDEF=SYSB==*========(00 BROWSE )====PLEXMGR==D====1
Cas Name........ C002      Description... TCP/IP Test CAS
Current System. No        Status........ Installed
Identification--
MVS Sysname.... C002      SMF ID........ *
Sysplex...... (*) Subsys Name... *
Communication--
VTAM Appl....... *NONE*    XCF Group..... C4000ALL
VTAM Mode...... *NONE*    IP Address.... 172.017.008.092
Port Number.... 4104      Host Name..... 172.017.008.092
TCPIP Jobname.. Parmlib Suffixes
SepDef Suffix.. 00        TgtDef Suffix. 00
SecDef Suffix.. 00        ConDef Suffix. 00
Common SerDef.. 00        Security Info---
Appl ID........          Logger SSID...
Update Info-----
UpdSystem....... SYSC    UpdTime....... hh:mm:ss
UpdUser......... MVQ10    UpdDate....... ddmmmyyyy
Deleted?....... No

For information about any field on the CASDEFD view, place the cursor on the field and press the Help key.

Editing CAS definitions

To obtain the edit lock on the CAS definition member (BBMTYB00) in the UBBPARM library and use other commands available in the CASDEF or CASDEFD views, use the EDIT command.

You cannot add, change, or delete CAS definitions unless you first obtain an edit lock on a CAS definition member.

When the edit lock is obtained, the edit mode status field on the window information line changes from (00 BROWSE) to (00 EDIT) to indicate that the edit lock is active.

The EDIT command is entered on the COMMAND line of the CASDEF or CASDEFD view.

When you are in edit mode, you can:

- Add CAS definitions
- Change CAS definitions
- Cancel CAS definition edits
- Delete CAS definitions
- Recover deleted CAS definitions
- Permanently save CAS definition edits

Adding CAS definitions

To add new CAS definitions, use the ADD command.
You must be in edit mode before you can add a new definition. See “Editing CAS definitions” on page 102 for more information.

When the ADD command is used, the Add CAS System Definition dialog is displayed with default values or with the values of an existing CAS definition that you want to use as a template for the new definition.

To save a new CAS definition (BBMTYB00) in the UBBPARM parameter library, use a SAVE or END command (or its corresponding PF key) as described in “Saving CAS definitions” on page 109.

To cancel a new CAS definition while you are defining it, use the CANcel command as described in “Canceling CAS definition edits” on page 106.

The following forms of the ADD command are available:

- **Primary command**
  
  From the CASDEF view, the primary command is entered on the COMMAND line as follows:
  
  `ADD casname`
  
  where `casname` is the name of the CAS definition that you want to create. This name should be the same as the system where the CAS definition is being created (unless you are defining a test CAS and a production CAS is already running on the same system).
  
  The Add CAS System Definition dialog is displayed with the default values except for the **CAS System Name** and **MVS System Name** fields. They contain the `casname` value that you specified.
  
  From the CASDEFD view, the primary command is entered on the COMMAND line as follows:
  
  `ADD`
  
  Because the CASDEFD view displays only one CAS definition at a time, the current definition is assumed to be the definition that you want to use as the template for a new definition.
  
  The Add CAS System Definition dialog is displayed with the values for the current definition. You must change at least the **CAS System Name** field.

- **Line command**
  
  The A line command is entered in the line command field of the CASDEF or CASDEFD view. The action is taken against the CAS definition where the command is entered.
  
  The current definition is assumed to be the definition that you want to use as the template for a new definition. The Add CAS System Definition dialog is displayed with the values for the current definition. You must change at least the **CAS System Name** field.
Changing existing CAS definitions

To change an existing CAS definition, use the CHAnge command in edit mode.

The Change CAS System Definition dialog is displayed, as shown in Figure 11 on page 104, with the details of the specified CAS definition. You can modify any field in this dialog except the CAS System Name field. Press the Help key for information about each field in this dialog.

Figure 11: Change a CAS System Definition dialog

```plaintext
| Change a CAS System Definition |
| COMMAND ===> |
| SCROLL ===> |
| PAGE |
| MORE: + |
| CAS System Name C002 (Recommend same as MVS System Name) |
| Description TCP/IP Test CAS |
| System Identification Information: |
| MVS System Name C002 * |
| SysPlex Name * Subsystem ID * |
| System Communication Information: |
| VTAM Appl Name |
| XCF Group Name C400QALL |
| IP Host Name 172.017.008.093 |
| Port Number 4104 TCP Jobname |
| ParmLib Suffix Information: |
| SepDef Suffix 00 TgtDef Suffix 00 |
| SecDef Suffix 00 ConDef Suffix 00 |
| Common SerDef 00 |
| Miscellaneous CAS Parameters |
| Logger SSID Override Security Applid |

END to exit saving changes
```

To save your changes as a member of the UBBPARM parameter library, use a SAVE or END command (or its corresponding PF key) as described in “Saving CAS definitions” on page 109.

To cancel your changes, use the CANcel command as described in “Canceling CAS definition edits” on page 106.

The following forms of the CHAnge command are available:

- Primary command

  From the CASDEF view, the primary command is entered on the COMMAND line as follows:

  ```plaintext
  CHA casname
  ```

  where `casname` is the name of the CAS definition that you want to change.

  From the CASDEFD view, the primary command is entered on the COMMAND line as follows:

  ```plaintext
  CHA
  ```

  Because the CASDEFD view displays only one CAS definition at a time, the current definition is assumed to be the definition that you want to change.
Line command

The C line command is entered in the line command field of the CASDEF or CASDEFD view. The action is taken against the CAS definition where the command is entered.

Defining TCP/IP communication

If you want to use TCP/IP for CAS-to-CAS communication, make sure that you are aware of the restrictions on specifying SUB=MSTR on the CAS START command.

For information about the restrictions on specifying SUB=MSTR on the CAS START command, see the table in “Starting a CAS” on page 21.

To define TCP/IP communication for a CAS, use the following fields on the Add or Change CAS System Definition dialog:

IP Hostname

Specify a unique host name that resolves to an IP address or the IP address itself. An IP address consists of four decimal integers between 0 and 255, with each number separated by a period. The first integer must be less than 224.

Port Number

Specify a unique port number for the CAS as an integer between 1024 and 65535.

Note

Be sure to assign a unique port number to each application that runs on a given system image. For example, the CAS must use a different port number than the MainView host server. The default port number for the MainView host server is 3940.

TCP Jobname

(optional) If you are running multiple TCP/IP stacks and you want MainView products to use a specific stack, specify the job name of the TCP/IP stack to be used. Make sure that the host name or IP address that you specified in the IP Hostname field is served by this stack.

In addition to specifying a host name or IP address, a port number, and, optionally, a TCP/IP job name, you must consider the following requirements for TCP/IP communication:

- Secure TCP/IP ports
- OMVS segment definitions
Secure TCP/IP ports

BMC recommends that you use a secure TCP/IP port for each production CAS. This topic explains how to secure a TCP/IP port for use by a CAS.

The profile data set for your TCP/IP started task, contains a section for reserving the port. If the started task name and step name of your CAS are the same, you can reserve the port number by started task name. If you use a different step name, you must reserve the port number by the step name of your CAS started task.

The following example is from SYSI.IBMTCP.PRD.CNTL(PROFILE):

```
:Reserve ports for the following servers.

NOTES:

A port that is not reserved in this list can be used by any user.
If you have TCP/IP hosts in your network that reserve ports in the range 1 - 1023 for privileged applications, you should reserve them here to prevent users from using them.

The port values below are from RCF 1060, "Assigned Numbers."

PORT 4100 TCP C410 ; CAS 41 reserved PORT
```

OMVS segment definitions

MainView TCP/IP communication makes use of z/OS UNIX System Services. A z/OS UNIX security context, called an OMVS segment, is required for any user ID that requests these services.

For more information about OMVS segment definitions, see “OMVS segment requirements and ESM definitions” on page 331.

Canceling CAS definition edits

To exit edit mode, the Add CAS System Definition dialog, or the Change CAS System Definition dialog, use the CANcel command.

You can cancel your changes at any time while in edit mode.

If you enter CANcel when the Add or Change dialogs are displayed, you are returned to the CASDEF or CASDEFD view, and the edit lock is still active.

If you enter CANcel when either the CASDEF or CASDEFD view is displayed with the edit lock active, you are returned to browse mode. The UBBPARM CAS definition parameter member is restored to its original contents when the edit lock was obtained or since the last SAVE or END command was used. The edit mode status field on the window information line changes from (00 EDIT) or (00 EDIT MOD) to (00 BROWSE) to indicate that the edit lock is no longer active.
Deleting a CAS definition

To delete specific CAS definitions from the UBBPARM parameter library, use the DELete command in edit mode.

A CAS definition is marked for deletion and is removed from display when the DELete command is entered. However, the definition is not actually removed from the UBBPARM CAS definition member (BBMTYB00) until a SAVE or END command (or its corresponding PF key) is used (see “Saving CAS definitions” on page 109 for more information). Until the change is saved, you can UNDelete a CAS definition to recover it as described in “Recovering a CAS definition” on page 107.

The following forms of the DELete command are available:

- Primary command
  
  From the CASDEF view, the primary command is entered on the COMMAND line as follows:

  \[ \text{DEL casname} \]

  where \( \text{casname} \) is the name of the CAS definition that you want to delete.

  From the CASDEFD view, the primary command is entered on the COMMAND line as follows:

  \[ \text{DEL} \]

  Because the CASDEFD view displays only one CAS definition at a time, the current definition is assumed to be the definition that you want to delete.

- Line command
  
  The DEL line command is entered in the line command field of the CASDEF or CASDEFD view. The action is taken against the CAS definition where the command is entered.

Recovering a CAS definition

To recover a CAS definition that was deleted with the DELete command, use the UNDelete command in edit mode.

The UNDelete command can recover a definition that was deleted as long as a SAVE or END command (or its corresponding PF key) has not been used.

A CAS definition can be redisplayed with UNDelete or by entering the following command on the COMMAND line:

\[ \text{PARm DELETED(*)} \]

This command displays all of the CAS definitions that are currently marked for deletion.
Undeleting a CAS definition also causes the **Del** field value in the CASDEF view or the **Deleted?** field value in the CASDEFD view to change from No to Yes.

The following forms of the UNDelete command are available:

- **Primary command**
  
  From the CASDEF view, the primary command is entered on the **COMMAND** line as follows:

  `UND casname`

  where `casname` is the name of the CAS definition that you want to recover. If you do not remember the name of the CAS, use the PARm DELETED(*) command first.

  From the CASDEFD view, the primary command is entered on the **COMMAND** line as follows:

  `UND`

  Because the CASDEFD view displays only one CAS definition at a time, the current definition is assumed to be the definition that you want to recover.

- **Line command**
  
  The UND line command is entered in the line command field of the CASDEF or CASDEFD view. The action is taken against the CAS definition where the command is entered.

**Immediately installing changes to CAS definitions**

To immediately install changes to CAS definitions without waiting until the CAS is recycled or the next system IPL, use the INStall command.

The INStall command dynamically updates the runtime version of a CAS definition.

---

**Note**

INStall *does not* update identification or cross-system communication parameters in CAS definitions and *does not* save changes to the UBBPARM member. A SAVE or END command (or its corresponding PF key) updates the UBBPARM member (see “Saving CAS definitions” on page 109 for more information).

---

The following forms of the INStall command are available:
Primary command

From the CASDEF view, the primary command is entered on the COMMAND line as follows:

**INS casname**

where *casname* is the name of the CAS definition that you want to update dynamically.

From the CASDEFD view, the primary command is entered on the COMMAND line as follows:

**INS**

Because the CASDEFD view displays only one CAS definition at a time, the current definition is assumed to be the definition that you want to update.

Line command

The INS line command is entered in the line command field of the CASDEF or CASDEFD view. The action is taken against the CAS definition where the command is entered.

Saving CAS definitions

To save a new CAS definition as a member of the UBBPARM library, or to save changes to existing CAS definitions in the UBBPARM library, enter the SAVE command.

SAVE can be used at any time in edit mode. The edit mode status field on the window information line changes from *(00 EDIT MOD)* to *(00 EDIT)* to indicate that the edit lock is still active and all modifications are saved.

*Note*

The END command (or its corresponding PF key) also saves any changes that you made while in edit mode.

Deploying CAS definitions

With CAS definition deployment, you can deploy one or more CAS definitions from one file system to one or more file systems for use on other systems.

You can use MainView Explorer or the MainView windows environment to deploy CAS definitions.

To deploy definitions

1. Display the CASDEF view.
2 Select one or more definitions that you want to deploy.

- MainView windows environment
  Enter the Tag line command next to one or more definition names. Use the Tag line command to clear a selection.

- MainView Explorer
  —To select one definition, click a definition name.
  —To select multiple definitions, click the Enable Selections button, press and hold the Ctrl key, and click the definition names.

3 Start the deployment wizard.

- In MainView windows, enter the DEPloy primary command.

- In MainView Explorer, right-click a selected definition name and choose Line action => Deploy Definition.

4 Complete the deployment wizard.

After completing the deployment wizard, the deployment process starts and the DPLYMNT view is displayed showing the status of the request. For more information about the DPLYMNT view, see “Managing and monitoring the deployment process” on page 110.

Managing and monitoring the deployment process

Use the DPLYMNT view to monitor and manage the deployment process. All deployment processes are monitored from the DPLYMNT view, including CAS definition deployment. The DPLYMNT view lists:

- All the deployment requests that have been made from the current system
- The status of the deployment requests

From the DPLYMNT view, you can:

- Stop, start, delete, and cancel deployment requests by using the available line commands
- See details about a deployment request by using the Select line command
- See details about the status of a deployment request by hyperlinking from the status
Displaying active systems and sessions

The Plex Manager CASACT view shows the status of all CAS-to-CAS communication links.

The CASACT view also enables you to manage cross-system VTAM communication links between CASs. By entering commands from this view, you can start, halt, or quiesce cross-system VTAM communication links between CASs.

---

Note

Commands associated with the CASACT view affect the current CAS only, which is identified by a Yes in the Cur Sys field.

To access the CASACT view

- In Plex Manager, type CASACT on the COMMAND line and press Enter.
- From any MainView product that is running in windows mode, use the following CONtext command to access Plex Manager and display the CASACT view:

  \[\text{CON} * \text{PLEXMGR;CASACT}\]

The CASACT view is displayed, as shown in Figure 12 on page 111.

---

Figure 12: CASACT view (left)

You can scroll to the right to display the remaining fields of the CASACT view, as shown in Figure 13 on page 111.

---

Figure 13: CASACT view (right)

For information about any field in the CASACT view, place the cursor on the field and press the Help key.
Viewing information about your current CAS

The CASINFO view provides information about the CAS you are running on, including the version and PUT level of the CAS.

To access the CASINFO view

1. Perform one of the following actions:
   - Type `CASINFO` on the command line of a view and press Enter.
   - Select CAS Information from the EZPLEX view.

The CASINFO view is displayed, as shown in Figure 14 on page 112.

--- Note ---
If the CASINFO view is invoked under SSI mode and some of the targets in the SSI do not have the PTF installed, the following error exceptions can be returned from those targets:

```plaintext
BBMYX335I Message(s) received from targets on target_name

- Related:BBMXB06E Failed to allocate a Join Object for Query.
  - Related:BBMXC01E The Allocate Function for Join returned - RC: 8
- Related:BBMXC01E The Allocate Function for DataRecord returned - RC: 8
- Related:BBMXC01E The Allocate Function for DataManagerStep returned - RC: 8
  - Related:BBMXC01E The Allocate Function for DataRecord returned - RC: 8
- Related:BBMXC01E The Allocate Function for Selector returned - RC: 8
  - Related:BBMXC01E The Allocate Function for DataRecord returned - RC: 8
```

---

Figure 14: CASINFO view (left)

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) ----------------
COMMAND ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
+W1 =CASINFO===========CXTST2===*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D====1
CAS      MVS      CAS  MVLO MVI     Maint   PUT      A COMM     M Exte XCF
Name---- Name---- SSID SSID Release Level-- Level--- M Order--- - Secu Group--
SJSEC610 SJSE     C610 MVSE V6.1.00 BPY9343 PUT1201B Y TXV      Y RACF MV61GRP
```

You can scroll to the right to display the remaining fields of the CASINFO view, as shown in Figure 15 on page 112 and Figure 16 on page 112.

Figure 15: CASINFO view (middle)

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) ----------------
COMMAND ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
+W1 =CASINFO===========CXTST2===*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D====1
CAS      XCF      UIM      UIM   UIM                              P T HostServ
Name---- Name---- SSID SSID Release Level-- Level--- M Order--- - Secu Group--
SJSEC610 MV61GRP  MV61PUIM 03661 V05.04.00(Build                  Y Y 172.24.4
```

Figure 16: CASINFO view (right)

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) ----------------
COMMAND ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
+W1 =CASINFO===========CXTST2===*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D====1
CAS      HostServer       HostS Path
```

Managing CAS-to-CAS communication

MainView Administration Guide
Starting VTAM communication

The STart command establishes VTAM communication with a remote CAS that is newly defined, quiesced, or has been previously halted.

**Note**
The STart command has no effect on the local CAS nor on any CAS that has an active communication link (VTAM Act field is Yes).

After you enter the STart command, results are as follows:

- The following message is displayed:
  
  BBMYAD101 START command accepted; CAS communications will start on next cycle

- The VTAM communication link between the local CAS and a remote CAS becomes active.

- The VTAM Act field of the CASACT view changes from No to Yes.

**Note**
The STart command operates on a timer cycle. When you issue the STart command the request might take as much as 10 minutes to complete. This delay is designed to limit CPU overhead.

The following forms of the STart command are available:

- **Primary command**
  
  The primary command is entered on the COMMAND line as follows:

  \[ ST \text{ } system \]

  where \textit{system} is the name of a remote CAS listed in the System field.

- **Line command**
  
  The ST line command is entered in the line command field (CMD) at the left of the CASACT view. VTAM communication is started for the remote CAS where the command is entered.
Stopping VTAM communication

Two methods are available for stopping VTAM communication.

When you quiesce communication with a CAS, VTAM communication does not terminate until all outstanding information requests are completed between CASs. By contrast, when you halt VTAM communication between CASs, VTAM communication terminates immediately, regardless of outstanding requests for information.

Quiescing VTAM communication

The QUIEsce command inactivates VTAM communication between a local and a remote CAS.

When you quiesce communication, all VTAM communication stops between a local and remote CAS after processing all pending requests for information. For example, suppose your local CAS is SYSA. You open a window, access MainView for z/OS on SYSB, and enter the WFLOW command. If communication between SYSA and SYSB is quiesced, the VTAM communication link does not end until WFLOW is displayed.

After VTAM communication quiesces, you cannot access the remote CAS until you restart communication again with the STart command; see “Starting VTAM communication” on page 113 for more information.

Note
Before using the QUIEsce command, you should display the DIAGSESS view to see a list of the active product and user sessions communicating with the local CAS. Based on the information shown in DIAGSESS, you can determine the potential impact of quiescing CAS communication.

After you enter the QUIEsce command, the following conditions exist:

- All pending requests for information are processed.
- Windows opened to the remote CAS are terminated for terminal sessions logged on through the local CAS.
- Windows opened to the local CAS are terminated for terminal sessions logged on through the remote CAS.
- VTAM links between the two CASs become inactive.
- Windows eventually become empty and a T appears in the window status field on the window information line.
Note
You cannot use the QUIEsce command against the local CAS, which is identified by a Yes in the Cur Sys field of the CASACT view.

The following forms of the QUIEsce command are available:

- Primary command
  The primary command is entered on the COMMAND line as follows:
  
  QUIE system

  where system is the name of a remote CAS listed in the System field.

- Line command
  The QUI line command is entered in the line command field (CMD) at the left of the CASACT view. VTAM communication is quiesced for the remote CAS where the command is entered.

Halting VTAM communication

The HALt command stops VTAM communication between a local and a remote CAS.

All VTAM communication between the local and remote CAS terminates immediately. After the communication link is inactive, you must use the STart command to reestablish the link to another CAS. See “Starting VTAM communication” on page 113 for more information.

Note
You should display the DIAGSESS view to see a list of active product and user sessions communicating with the local CAS before using the HALt command. Based on the information presented by the DIAGSESS view, you can determine the potential impact of halting communication between CASs.

After you enter the HALt command, the following conditions exist:

- Windows opened to the remote CAS are terminated for terminal sessions logged on through the local CAS.

- Windows opened to the local CAS are terminated for terminal sessions logged on through the remote CAS.

- VTAM links between the local and remote CASs become inactive.

- Windows are blank and a T appears in the window status field on the window information line.
Note
You cannot use the HALt command against a local CAS, which is identified by a Yes in the Cur Sys field of the CASACT view.

The following forms of the HALt command are available:

- Primary command
  The primary command is entered on the COMMAND line as follows:
  \texttt{HAL system}
  where \textit{system} is the name of a remote CAS listed in the System field.

- Line command
  The HAL line command is entered in the line command field (CMD) at the left of the CASACT view. VTAM communication stops for the remote CAS where the command is entered.

Managing cross-system communication between BBI-SS PASs

This topic describes how to manage VTAM communication between BBI-SS PASs located on different system images.

Tip
The information in this topic applies to the following products only:

- MainView AutoOPERATOR
- MainView FOCAL POINT
- MainView \textit{for} CICS
- MainView \textit{for} DB2
- MainView \textit{for} DBCTL
- MainView \textit{for} IMS Online
- MainView \textit{for} MQ

The figure in “Managing cross-system communication” on page 97 illustrates VTAM links that enable MainView products in a BBI-SS PAS on one system image to communicate with products on another system image.
To start, stop, and display communication links between BBI-SS PASs, issue one of the following BBI control commands from a terminal session, a MainView AutoOPERATOR EXEC, or an operator's console:

.START LINK
.STOP LINK
.DISPLAY LINK

Table 12 on page 117 summarizes the use of these commands.

Table 12: Control commands for VTAM links between BBI-SS PASs

<table>
<thead>
<tr>
<th>Command</th>
<th>Keywords</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISPLAY</td>
<td>D</td>
<td>LINK</td>
<td>L</td>
</tr>
<tr>
<td>START</td>
<td>S</td>
<td>LINK</td>
<td>L</td>
</tr>
<tr>
<td>STOP</td>
<td>P</td>
<td>LINK</td>
<td>L</td>
</tr>
</tbody>
</table>

To use these BBI control commands, use the following guidelines:

- For command notation shown in Table 12 on page 117, conventions are as follows:
  - Command and keyword abbreviations are shown as an alternate choice as indicated by a vertical line between them.
  - The symbolic word, ssid, represents a one- to four-character BBI subsystem ID.

- For command syntax, use the following conventions:
  - Use a blank between commands and keywords.
  - Use an equal sign, blank, or comma between keywords and parameters.

See “BBI control commands” on page 77 for further information about the use of BBI control commands.
Managing MainView Explorer

Tip
The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

This chapter explains how to perform the following tasks:

- “Starting and stopping the MainView host server” on page 119
- “Setting MainView Explorer global options” on page 120

Starting and stopping the MainView host server

This section describes how to start and stop the MainView host server.

Starting the host server

Before users can launch MainView Explorer to access host data, the MainView host server must be running as a started task.

The host server must have an OMVS segment defined to the user that is associated with the authorized started task.

**To start the host server**

1. At the system console, type the following command:

   START procedure
where *procedure* is the catalogued procedure that contains the JCL to run the host server. The distributed procedure name is BBMXPJCL, however, it might have been renamed during installation.

After the host server initializes, the following messages are written to the system console:

BBWIA001I Explorer Host Server is starting
BBWIB030I CAS(LG88) connection is established
BBWIE001I MainView EXPLORER (v.r.mm) SERVICES ARE AVAILABLE

**Note**

- For information about creating the MainView host server startup procedure, see the *MainView Customization Reference*.
- For information about starting the MainView Explorer client, see the *MainView User Guide*.

### Stopping the host server

To stop the MainView host server, issue one of the following commands from the system console or the COMMAND line in SDSF

- **STOP**  
  P *procedure*

- **MODIFY**  
  F *procedure*,STOP

*procedure* is the name of the host server started task procedure.

### Setting MainView Explorer global options

Various global options are available for administrators to control site-wide MainView Explorer usage.

You can control usage in:

- **MainView Explorer** - Global options are specified in member Z$MVE of the SBBCDEF site configuration library. SBBCDEF is created during customization and is allocated in the MainView host server JCL procedure.
MVE Viewer - Global options are specified in a file called Z$MVE that you create in the export directory where exported data files and configurations reside.

To set global options for MainView Explorer

1. Create member Z$MVE in the SBBCDEF library.

2. Add the global option statements that you want to use to Z$MVE, as described in the table in “Global option statements” on page 121.

3. Save the Z$MVE member.

To set global options for the MVE Viewer

1. Create a file called Z$MVE (no file extension) in the export directory that contains exported data files and configurations.

2. Add the global option statements that you want to use to Z$MVE, as described in the table in “Global option statements” on page 121.

   **Note**

   You can use the Z$MVE member in SBBCDEF as the basis for an MVE Viewer global options file. However, not all global options are supported by the MVE Viewer. For more information, see “Global option statements” on page 121.

3. Save the Z$MVE file.

   **Note**

   If you do not want MVE Viewer users to be able to change exported configurations in the export directory, make the directory read-only. If you specify EXPORTRO=Y in the Z$MVE file, MVE Viewer users will not see the Save Configuration menu option or toolbar button.

Global option statements

Use the statements in the following table to set MainView Explorer global options in the Z$MVE member or file.

The syntax is explained in “Rules for global option statements” on page 124.
### Table 13: Global options

<table>
<thead>
<tr>
<th>Statement</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPRESS=N</td>
<td>Prevents GZIP compression of export data</td>
</tr>
<tr>
<td></td>
<td>The default is Y. Exported view data files (.1 and .vdf) are compressed by default. The MVE Viewer can import either compressed or uncompressed data files. To turn off data compression at your site, specify COMPRESS=N.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Exported .csv files are never compressed. They contain only the visible columns in a view and can be opened directly in a spreadsheet application.</td>
</tr>
<tr>
<td>CONFIG=name</td>
<td>Sets a default configuration for first time users</td>
</tr>
<tr>
<td></td>
<td>If you specify a default configuration, make sure the member exists in the site configuration library (SBBCDEF). Users can override the site default by setting their own configuration, either in their user preferences, or by using the CONFIG parameter on the launching URL.</td>
</tr>
<tr>
<td>CONFIRM=Y</td>
<td>Displays a message before performing line actions that are marked for confirmation</td>
</tr>
<tr>
<td></td>
<td>The default is N. By default, MainView Explorer does not display a message before executing product line actions that are marked for confirmation. To ensure that a confirmation message is displayed, specify CONFIRM=Y.</td>
</tr>
<tr>
<td>DATEFORMAT=1</td>
<td>Sets the date format as one of the following:</td>
</tr>
<tr>
<td></td>
<td>■ 1 - MM/DD/YY</td>
</tr>
<tr>
<td></td>
<td>■ 2 - DD/MM/YY</td>
</tr>
<tr>
<td></td>
<td>■ 3 - YY/MM/DD</td>
</tr>
<tr>
<td></td>
<td>The default is 1.</td>
</tr>
<tr>
<td>EXPORTRO=N</td>
<td>Indicates the export directory is not read-only</td>
</tr>
<tr>
<td></td>
<td>The default is N. To make the export directory read-only, specify EXPORTRO=Y.</td>
</tr>
<tr>
<td>Statement</td>
<td>Function</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| NODE3270=NO      | Removes the 3270 node from the products tree, which disables the use of the MainView Explorer internal 3270 emulator  
Some views contain hyperlinks to ISPF-only views. When you click a hyperlink, the 3270 emulator window is opened automatically to provide access to those views. If you stop the use of the MainView Explorer 3270 emulator, and a 3270 emulator is needed to display the requested information, a message is displayed. The message instructs the user to open an emulator, access the MainView Selection Menu, and issue the supplied statement to see the requested view. The following dialog box is an example of the message: |
| NODEPLEX=NO      | Removes the PLEX node from the products tree  
To view the Plex map, click on its tab in the tabbed panes.  
The Plex Map shows the active MainView products, the systems on which they are active, and the connections between them. |
| NODETOPO=YES     | Includes Plex topology nodes in the products tree  
If you specify YES, two nodes appear in the products tree labeled Systems and Products. These nodes can be used to open and close the Systems Plex and Products Plex topology displays. The default is NO. |
| NODEWARN=NO      | Removes the Alerts node from the products tree  
The Alerts view can still be accessed from the MVALELT product. |
| PLEXMAP=NO       | Suppresses the Plex Map display  
The default is YES. |
| PORT3270=nn      | Changes the default TN3270 port number  
The default is 23. |
| QLIMIT=nn        | Sets the number of query views you can have open at one time  
When this value is reached, you are warned about having too many query views open. The default is 30. |
| REFRESH=nn       | Sets the minimum auto-refresh interval to the number of seconds specified  
The default is 15 seconds. |
| REFRESH=NO       | Disables auto-refresh completely |
| SHOWDATE=Y a     | Displays the DATE column in Time duration mode  
The default is Y. To hide the DATE column, specify SHOWDATE=N. |
| Statement       | Function                                                                 \ 
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SHOWSYSTEM=Y</td>
<td>Displays the SSI SYSTEM column in SSI mode</td>
</tr>
<tr>
<td></td>
<td>The default is Y. To hide the SSI SYSTEM column, specify SHOWSYSTEM=N.</td>
</tr>
<tr>
<td>SHOWTARGET=Y</td>
<td>Displays the SSI TARGET column in SSI mode</td>
</tr>
<tr>
<td></td>
<td>The default is Y. To hide the SSI TARGET column, specify SHOWTARGET=N.</td>
</tr>
<tr>
<td>SHOWTIME=Y</td>
<td>Displays the TIME column in Time duration mode</td>
</tr>
<tr>
<td></td>
<td>The default is Y. To hide the TIME column, specify SHOWTIME=N.</td>
</tr>
<tr>
<td>TITLE='text'</td>
<td>Replaces BMC MainView Explorer in the title of application’s main window with the specified text</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This option applies only when MainView Explorer is run as a local application, not when it is launched in a web browser.</td>
</tr>
<tr>
<td>TIMEOUT=nn</td>
<td>Sets the timeout for query views to the number of seconds specified</td>
</tr>
<tr>
<td></td>
<td>The default is 30 seconds.</td>
</tr>
<tr>
<td>TOPOPROD=NO</td>
<td>Suppresses the Products Plex Topology display</td>
</tr>
<tr>
<td></td>
<td>If TOPOPROD is missing from member Z$MVE (or you specify YES), the Products Plex Topology is launched as usual.</td>
</tr>
<tr>
<td>TOPOSYST=NO</td>
<td>Suppresses the Systems Plex Topology display</td>
</tr>
<tr>
<td></td>
<td>If TOPOSYST is missing from member Z$MVE (or you specify YES), the Systems Plex Topology is launched as usual.</td>
</tr>
</tbody>
</table>

a These options are the only global options that are supported by the MVE Viewer.

### Rules for global option statements

The following rules apply to global option statements:

- The option statements consists of a keyword, an equal sign (=), and a value.
- An asterisk in column 1 denotes a comment and is ignored, as is any text following the asterisk.
- Keywords and values are not case sensitive.
- Blank spaces in the option statements are ignored.
- Text following a value is ignored.
- Blank lines in the Z$MVE member are ignored.
- If duplicate statements exist in the Z$MVE member, the last statement encountered is used.
If a value is surrounded by quotation marks (single or double), the value is case-sensitive and blank spaces are recognized, such as the following statement:

TITLE='ABC Company'
Managing MainView Logger

This chapter describes the MainView Logger and explains how to manage the collection, storage, retention, and retrieval of information from MainView products.

Tip

The information described in this chapter applies to all MainView products.

Understanding MainView Logger

MainView Logger is a stand-alone logging facility that provides any BMC product or MainView Infrastructure a way to log important messages or events in a centralized repository.

The MainView Logger agent runs within the DB2 Component Services (DBC) subsystem address space. For more information about the DBC address space, see “Working with DB2 Component Services (DBC)” on page 305. The MainView Logger agent manages the logspace and provides connections that supply MainView Logger functionality.

Securing MainView Logger

There are no additional security considerations for the MainView Logger because it runs as a service in the DBC address space. The DBC address space on each system requires ALTER access to the logset data sets that MainView products write their messages to, just as it requires ALTER access to logset data sets for other products. End-users and MainView started tasks do not directly access the MainView Logger logset data sets so there are no additional security considerations.

For more information, see “DBC security” on page 320.

Logspaces

Each instance of the MainView Logger agent can support multiple logspaces. Each logspace requires its own Next Generation Logger (NGL) logset which also runs as
an agent within the DBC. There is a one-to-one relationship between a logspace and its corresponding NGL logset. In addition:

- The default logspace is used by MainView Infrastructure logspaces and (optionally) by other products. This default logspace is also called the primary logspace and is usually where WTO messages are written.

- The product-specific logspaces are used when products have special data requirements or as a way to organize large numbers of log entries. Product-specific logspaces are installed after installing the primary logspace.

- You cannot dynamically start and stop a logspaces however, you can dynamically start and stop the NGL logset that maps to the logspace with NGL commands. For more information, see “Controlling MainView Logger operation” on page 146.

The NGL logset data sets are created when the MainView Logger agent initializes in the DBC. During initialization, the MainView Logger agent reads the specifications in BBPARM member LOGGERxx to define the NGL logsets. For additional information about the DBC and NGL, see the BMC Infrastructure Components Administration Guide.

**Definition members**

BBPARM member LOGGERxx stores parameters that identify and define the logspace. Each logspace must have it's own BBPARM member LOGGERxx.

You can use the sample member BBSAMP(NGLOGGER) as a model to define a LOGGERxx member. Then, specify the two-character BBPARM(LOGGERxx) suffixes (up to ten) on the SYSP parameter of BBPARM member MVINGIxx. To define the default logspace, specify DEFAULTLOGGER=YES in the LOGGERxx member for the default logspace. The MainView Logger agent reads the entries from all of the LOGGERxx members specified in BBPARM member MVINGIxx and uses the LOGGERxx values to start the logspaces.

When the MainView Logger agent starts, it does the following:

- Reads the LOGGERxx definition members specified with the SYSP parameter in BBPARM member MVINGIxx.
  
  For example, you might have these three LOGGERxx members suffixes specified in BBPARM member MVINGIxx: SYSP=(00, AD, 12).

  When a single MainView Logger agent manages multiple logspaces, you must specify all of the suffixes (up to 10) from the associated LOGGERxx definition members on the SYSP parameter in BBPARM member MVINGIxx. Also, specify a unique name for each logspace with the LOGSPACE parameter in each LOGGERxx member.
The MainView Logger agent recognizes only one logspace with the DEFAULTLOGGER=YES parameter per system image. Any subsequent logspace attempting to start as the default logger will fail.

- Locates the system entries in the LOGGERxx members.
- Uses the parameter values from the LOGGERxx member for that system to start the logspaces.

**Started task procedures**

Each system image that uses logspaces must have a MainView Logger agent running within the image.

A started task procedure is required to start each instance of MainView Logger agent and its associated default logspace. You can start additional logspaces by using the existing NGL started task procedure or by creating a separate procedure for each logspace.

*Tip*

BMC recommends that on a specific image, you start all logspaces from one NGL instance.

For more information about manually configuring product-specific logspaces, see the *MainView Customization Reference*.

**Managing MainView Logger definition members**

BBPARM members LOGGERxx and MVINGIxx contain parameters that define MainView Logger logspaces.

The default BBPARM members LOGGERxx and MVINGIxx are created during MainView Infrastructure customization. You can also create these members manually from BBSAMP members NGLOGGER and MVINGINI. For more information about manually creating and editing the parameters in these members, see the *MainView Customization Reference*.
**Note**

- Any additions, deletions, or changes made to parameters settings in BBPARM LOGGERxx or MVINGIxx members do not take affect until you restart the NGL logset for the logspace.

- When you add or delete BBPARM LOGGERxx members, you must also update LOGGERxx suffixes specified on the SYSP parameter in the BBPARM member MVINGIxx to add or delete the LOGGERxx members.

The following topics describe how to manage BBPARM LOGGERxx members and the definitions that they contain.

### Displaying a list of logger definition members

You can use the LOGRDEFL view to display a list of all the MainView Logger definition members (LOGGERxx) in the BBPARM parameter library.

You can also add, delete, or change a definition member.

**To access the LOGRDEFL view**

1. Perform one of the following actions:
   - In Plex Manager, type `LOGRDEFL` on the **COMMAND** line and press **Enter**.
   - From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the LOGRDEFL view:
     ```
     CON * PLEXMGR;LOGRDEFL
     ```

LOGRDEFL is displayed in a window.

**Figure 17: LOGRDEFL view**

<table>
<thead>
<tr>
<th>dmmmmyyyy hh:mm:ss</th>
<th>MainView WINDOW INTERFACE (Vv..mm)</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURR WIN</td>
<td>1</td>
<td>ALT WIN</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

For more information about any field in the LOGRDEFL view, place the cursor on the field and press the **Help** key.

**To manage MainView Logger definition members, use the action commands listed in the following table.**
Table 14: Logger definition member tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a new logger definition member to the BBPARM parameter library</td>
<td>ADD primary command or A line command</td>
<td>“Adding a logger definition member” on page 131</td>
</tr>
<tr>
<td>Delete a logger definition member from the BBPARM parameter library</td>
<td>DELete primary command or DEL line command</td>
<td>“Deleting a logger definition member” on page 132</td>
</tr>
<tr>
<td>Edit an existing logger definition member</td>
<td>EDIT primary command or E line command</td>
<td>“Editing a logger definition member” on page 131</td>
</tr>
<tr>
<td>Browse a logger definition</td>
<td>BROwse primary command</td>
<td>“Managing entries in a logger definition member” on page 132</td>
</tr>
</tbody>
</table>

Adding a logger definition member

To add a MainView Logger logspace definition member to the BBPARM parameter library, perform one of the following actions:

- On the COMMAND line, type ADD xx, where xx is the suffix of the new member that you want to add. The NGLOGGER member is used as a model for the new member.

- In the CMD field, type A next to a definition member that you want to use as a model and type over the ID field with a unique member suffix.

After you press Enter, the following message is displayed:

```
BBMYC123I Parameter member LOGGERxx is missing - press ENTER to use default
```

This message indicates that the new member is empty except for the default definitions. When you press Enter again, the LOGRDEF view is displayed in Edit mode. To manage the information in the new member, use the action commands shown in the table in “Managing entries in a logger definition member” on page 132.

Editing a logger definition member

To select a MainView Logger logspace definition member for editing, perform one of the following actions:
On the COMMAND line, type EDIT xx, where xx is the suffix of the member that you want to edit.

In the CMD field, type E next to the definition member that you want to edit.

After you press Enter, the LOGRDEF view is displayed in Edit mode. To manage the definitions in the selected member, use the action commands shown in the table in “Managing entries in a logger definition member” on page 132.

Deleting a logger definition member

To delete a MainView Logger definition member from the BBPARM parameter library, perform one of the following actions:

- On the COMMAND line, type DELete xx, where xx is the suffix of the member that you want to delete.
- In the CMD field, type DEL next to the definition member that you want to delete.

**WARNING**

No confirmation panel is displayed before the definition member is deleted. Be sure that you want to delete the member.

After you press Enter, the member is deleted from BBPARM.

Managing entries in a logger definition member

You use the LOGRDEF view to display information about the system entries, in a given LOGGERxx member.

You can also add, delete, or change entries in the member.

**To access the LOGRDEF view**

1. Perform one of the following actions:

   - In the LOGRDEFL view, use the BROwse primary or B line command to display a specific logger definition member.

   - From elsewhere in Plex Manager, type LOGRDEF on the COMMAND line and press Enter.
From any MainView product that is running in windows mode, use the following CONtext command to access Plex Manager and display the LOGRDEF view:

```
CON * PLEXMGR;LOGRDEF
```

LOGRDEF is displayed in a window.

**Figure 18: LOGRDEF view**

```
ddmmmyyyy hh:mm:ss -------- MainView WINDOW INTERFACE (Vv.r.mm) ----------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
W1 =LOGRDEF==========MVSB=====*==========(00 BROWSE        )===PLEXMGR==D===2
CMD MVS Cur SSID Description WTOLOG MsgFilter MaxMsg Co
--- SYSNAME Sys ---- ----------- (Y/N) Suffix /Sec (Y
MVSB Yes SystemB Logger No 15 No
```

To display detailed information about a specific system entry, place the cursor on a name in the MVS SYSNAME field and press Enter.

The LOGRDEFD view is displayed.

**Figure 19: LOGRDEFD view**

```
ddmmmyyyy hh:mm:ss -------- MainView WINDOW INTERFACE (Vv.r.mm) ----------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
W1 =LOGRDEFD==========SJSC=====*========(00 BROWSE        )====PLEXMGR==D====1==
System Section----
MVS Name......... &SYSNAME   Description.... 00
Default Logger... NO         Mvlogger SSID.. LG21
Current SYstem... Yes        LOGSPACE.......
SYSP Parms..... 00
Operation Section-
Log WTO messages. YES
Msg filter suffix Register DSN... BB.XTST21.LG00.REGISTER
Max Svc Dump...... 1 Compress(Y/N).. YES
Multi Record...... YES Max msg(/Sec). 0
Logfile Section---
Storage Class.... Logfile prefix. BB.XTST21.LG00
Space Unit....... CYL Management Class
Space Quantities. 50
DIV Section------
DIV Blocks....... 180 DIV Dataset.... BB.XTST21.LG00.DIV
```

For more information about any field in the LOGRDEF or LOGRDEFD view, place the cursor on the field and press the Help key.

To manage the system entries in a logger definition member, use the action commands listed in the following table.

**Table 15: Logger definition tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obtain the edit lock on the logger definition member</td>
<td>EDIT primary command</td>
<td>“Editing system entries” on page 134</td>
</tr>
</tbody>
</table>
### Editing system entries

Before you can add, change, delete, recover, or save system entries in a logger definition member, you must first obtain the edit lock on the member.

To obtain the edit lock on a logger definition member, type **EDIT** on the **COMMAND** line and press **Enter**.

When the edit lock is obtained, the edit mode status field on the window information line changes from *(00 BROWSE)* to *(00 EDIT)* to indicate that the edit lock is active.

#### To add a system entry

1. On the **COMMAND** line, type **EDIT** and press **Enter** to obtain the edit lock.

2. Enter the **ADD** command in one of the following ways:

   - On the **COMMAND** line, type **ADD sysname**, where *sysname* is the unique name of a system where MainView Logger will run.
In the CMD field, type A next to a system entry that you want to use as a model.

3 Press Enter.

The Add a System Entry dialog is displayed.

**Figure 20: Add a System Entry dialog**

```
Add a System Entry in Parm Definition
COMMAND ====> SCROLL ====> PAGE
MORE: +
MVS System Name
Description
Mvlogger SSID Default Logger
LOGSPACE
Operation Section:
Register Dataset
Log WTO Messages Compress Logfile
Msg Filter member suffix Max Msg per second
Max Svc Dump to be taken Multi Record
Logfile Section:
Logfile Prefix
Storage Class Management Class
Log File Allocation: (Recommendation 50 Cyl)
Space Unit Cyl or Trk
Quantities In above units
DIV File Section:
DIV Dataset Name
Number of Blocks
```

4 Enter information for the new system entry in the Add a System Entry dialog.

For more information about the fields in this dialog, press the Help key.

5 Press the END key to return to the LOGRDEF or LOGRDEFD view.

**To change a system entry**

1 On the COMMAND line, type EDIT and press Enter to obtain the edit lock.

2 Enter the CHAnge command in one of the following ways:

- On the COMMAND line, type CHAnge sysname, where sysname is the name of the system entry that you want to modify.

- In the CMD field, type C next to the system entry that you want to modify.

3 Press Enter.

The Change MVS System Entry dialog is displayed. This dialog is similar to the Add a System Entry dialog.

4 Change the information for the system entry.
You can change any value except the **MVS System Name**. For more information about the fields in the Change MVS System Entry dialog, press the Help key.

5 Press the END key to return to the LOGRDEF or LOGRDEFD view.

**Deleting and recovering system entries**

When you delete a system entry, the entry is marked for deletion and removed from the LOGRDEF view.

However, the entry is not permanently removed from the LOGGER$xx$ member until the SAVE command is entered. A system entry can be recovered by using the UNDelete command before the definition member is saved.

**To delete a system entry from a logger definition member**

1 On the **COMMAND** line, type **EDIT** and press **Enter** to obtain the edit lock.

2 Enter the DELete command in one of the following ways:

   - On the **COMMAND** line, type **DELETE sysname**, where **sysname** is the name of the system entry that you want to delete.
   - In the **CMD** field, type **DEL** next to the system entry that you want to delete.

3 Press **Enter**.

   The system entry is marked for deletion and removed from the LOGRDEF view.

**To recover a previously deleted system entry**

1 On the **COMMAND** line, type **EDIT** and press **Enter** to obtain the edit lock.

2 On the **COMMAND** line, type **SHOWDel** and press **Enter**.

   All of the system entries that are currently marked for deletion are redisplayed in the LOGRDEF view.

3 Enter the UNDelete command in one of the following ways:

   - On the **COMMAND** line, type **UNDELETE sysname**, where **sysname** is the name of the system entry that you want to recover.
   - In the **CMD** field, type **UND** next to the system entry that you want to recover.

4 Press **Enter**.
The system entry is no longer marked for deletion.

**Controlling the display of deleted system entries**

By default, system entries that have been marked for deletion do not appear in the LOGRDEF view.

You can use the SHOWDel and HIDEdel commands to control the display of deleted entries.

To display system entries that are currently marked for deletion, type `SHOWDel` on the `COMMAND` line and press `Enter`. All of the deleted system entries are redisplayed in the LOGRDEF view.

To remove system entries that are marked for deletion from the LOGRDEF view, type `HIDEdel` on the `COMMAND` line and press `Enter`.

**Canceling system entry edits**

You can cancel your changes at any time during an edit session.

Use the CANcel command to exit:

- **Edit mode**
- **Add a System Entry dialog**
- **Change MVS System Entry dialog**

The CANcel command can be entered on the `COMMAND` line of the LOGRDEF and LOGRDEFD views or the Add or Change System Entry dialogs.

If you enter CANcel when either the Add or Change dialog is displayed, you are returned to the LOGRDEF or LOGRDEFD view. The changes that you were making are lost, but the edit lock remains active.

If you enter CANcel when either the LOGRDEF or LOGRDEFD view is displayed, the results are as follows:

- You are returned to browse mode; the edit mode status field on the window information line changes from `(00 EDIT)` or `(00 EDIT MOD)` to `(00 BROWSE)` to indicate that the edit lock is no longer active.
- The LOGGERxx member that you were editing is refreshed from storage with either the contents when the edit lock was obtained or the contents when the last SAVE or END command was entered.
Saving system entries

Use the SAVE command to save additions, changes, or deletions to a LOGGERxx member in the BBPARM parameter library.

The SAVE command is entered on the COMMAND line. You must be in edit mode before you can enter the SAVE command.

Note
The END command (or its corresponding PF key) also saves any changes that you made while in edit mode.

When the SAVE command is used, the Save MVLOGGER Parm Definition Member dialog is displayed.

Figure 21: Save MVLOGGER Parm Definition Member dialog

- If you are saving changes to an existing logger definition member, this dialog shows the previously defined Description field value.
- If you are saving a new logger definition member, this dialog is displayed with a blank Description field.

To save the logger definition member

1. Enter a description of up to 40 characters (it will appear in the Description field on the LOGRDEFL view).

2. Press END to save the description and any changes to the LOGGERxx member in the BBPARM library.

When you return to either the LOGRDEF or LOGRDEFD view, the edit mode status field on the window information line has changed from (00 EDIT MOD) to (00 EDIT) to indicate that the edit lock is still active but all modifications have been saved.
Managing the log files

MainView Logger logs use the NGL logset log file data sets to store messages and event records that are produced by MainView Infrastructure and other BMC products. The log files are owned and maintained by the NGL started task procedure. NGL contains information about the log files.

Displaying a list of log files

Use the LOGLIST view to review information about the log files that are used by the NGL MainView agent.

LOGLIST can help you determine:

- How many log files have been created
- How long a single log file lasts

To access the LOGLIST view

1. Perform one of the following actions:

   - In Plex Manager, type `LOGLIST` on the COMMAND line and press Enter.
   
   - From any MainView product that is running in windows mode, use the following CONtext command to access Plex Manager and display the LOGLIST view:

     `CON * PLEXMGR;LOGLIST`

     The LOGLIST view is displayed.

     **Figure 22: LOGLIST view**

     | Number | Time | Time | mm/dd/yyyy | Count | Block | Util % | Compress | Physical File Name |
     |--------|------|------|------------|-------|-------|--------|----------|-------------------|
     | NMVILG21 | 17:53 | 18:05 | 0035099 | | | 22.0% | BB.XTST21.NGLSJSC.C000001 |
     | NMVILG21 | 17:14 | 17:53 | 0124787 | | | 22.0% | BB.XTST21.NGLSJSC.C000002 |
     | NMVILG21 | 16:33 | 17:14 | 0125185 | | | 22.0% | BB.XTST21.NGLSJSC.C000003 |
     | NMVILG21 | 15:52 | 16:33 | 0125137 | | | 22.0% | BB.XTST21.NGLSJSC.C000004 |
     | NMVILG21 | 15:06 | 15:52 | 0125872 | | | 22.0% | BB.XTST21.NGLSJSC.C000005 |
     | NMVILG21 | 14:22 | 15:06 | 0125352 | | | 22.0% | BB.XTST21.NGLSJSC.C000006 |
     | NMVILG21 | 13:36 | 14:22 | 0125527 | | | 22.0% | BB.XTST21.NGLSJSC.C000007 |
     | NMVILG21 | 12:54 | 13:36 | 0125133 | | | 22.0% | BB.XTST21.NGLSJSC.C000008 |
     | NMVILG21 | 12:35 | 12:54 | 0126348 | | | 22.0% | BB.XTST21.NGLSJSC.C000009 |
     | NMVILG21 | 11:27 | 12:15 | 0125754 | | | 22.0% | BB.XTST21.NGLSJSC.C000010 |
     | QAUXT21 | 10:00 | 10:05 | 0120053 | | | 22.0% | BB.XTST21.AOAUSJSC.C000001 |
     | QAUXT21 | 09:45 | 10:00 | 0201154 | | | 22.0% | BB.XTST21.AOAUSJSC.C000002 |
     | QAUXT21 | 08:55 | 09:45 | 0197723 | | | 22.0% | BB.XTST21.AOAUSJSC.C000003 |
     | QAUXT21 | 07:57 | 08:55 | 0212798 | | | 22.0% | BB.XTST21.AOAUSJSC.C000004 |
     | QAUXT21 | 07:22 | 08:22 | 0201154 | | | 22.0% | BB.XTST21.AOAUSJSC.C000005 |
     | QAUXT21 | 03:49 | 04:37 | 0319537 | | | 22.0% | BB.XTST21.AOAUSJSC.C000006 |
     | QAUXT21 | 03:41 | 04:37 | 0208520 | | | 22.0% | BB.XTST21.AOAUSJSC.C000007 |
     | QAUXT21 | 02:44 | 03:41 | 0198046 | | | 22.0% | BB.XTST21.AOAUSJSC.C000008 |

Chapter 7 Managing MainView Logger 139
For more information about any field in the LOGLIST view, place the cursor on the field and press the Help key.

The LOGLIST view has no action commands.

Displaying logged messages

The following views are available to display the messages that were logged by MainView Logger:

- **LOGMSG**

  LOGMSG displays messages that were logged by MainView Logger, including extended diagnostic mode (XDM) messages, SYSLOG messages, and BBI-SS PAS journal records.

  To access the LOGMSG view, type `LOGMSG` on the **COMMAND** line and press **Enter**. LOGMSG is displayed.

- **LOGRPT**

  LOGRPT displays messages that were logged by MainView Logger, including extended diagnostic mode (XDM) messages, SYSLOG messages, and BBI-SS PAS journal records.

  The fields in the fixed section on the top of the LOGRPT view determines how many records are displayed. By default, the number of records requested for an individual target is 1000 and the time is set to the current time. Overtype the fields in the fixed section or use the LOGTIME primary command to request a different set of records to view.

  LOGRPT is a hybrid view that contains a fixed section and a scrollable section containing row data. The hyperlink on the first field of the fixed section (- | +) is used as a toggle to show (include) or hide (exclude) the display of the fixed section. The primary commands INC FIX and EXC FIX serve the same purpose as the - and + hyperlinks.

  To access the LOGRPT view, type `LOGRPT` on the **COMMAND** line and press **Enter**. LOGRPT is displayed.

- **LOGJRNL**

  LOGJRNL displays log entries specifically for MainView products that run in a BBI-SS PAS. The journal messages from a BBI-SS PAS are intermixed with all of the logged messages, making it difficult to use the view for BBI-SS PAS messages. The LOGJRNL view isolates the BBI-SS PAS journal messages and includes additional information to make the messages more useful, such as the point of entry of an event or message.

  To access the LOGJRNL view, type `LOGJRNL` on the **COMMAND** line and press **Enter**. LOGJRNL is displayed.
The LOGMSG, LOGRPT, and LOGJRNL views can be accessed from Plex Manager or any MainView product in either windows or full-screen mode. For products that are running in full-screen mode, LOGMSG, LOGRPT, and LOGJRNL are considered transfer commands rather than view commands. They transfer you from a full-screen application to the LOGMSG, LOGRPT, or LOGJRNL view in windows mode.

**Figure 23: LOGMSG view**

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) -------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
>W1 =LOGMSG=========MVSB======*=========ddmmmyyyy==hh:mm:ss====PLEXMGR==D===13
C # Log Log Jobname Log
- - Date------ Time-------- -------- Message---
2005/05/27 08:51:11.489 MVLOGGER BBMZL010I MVLOGGER INITIALIZING, VERSION-
2005/05/27 08:51:11.497 MVLOGGER BBMZL020I USING PARM MEMBER LOGGERROO
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I MVSYS=MVSB
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I LOGSSID=MVL1
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I MSGFLT=00
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I MAXPERSEC=15
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I DUMPCOUNTMAX=10
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I LOGCYLINDERS=10
2005/05/27 08:51:11.497 MVLOGGER BBMZL001I LOGPREFIX=BMC.MVLOGGER.MVL1MVSB
2005/05/27 08:51:11.524 MVLOGGER BBMZL051I LOG FILE 10326104 HAS BEEN RE-O
2005/05/27 08:51:11.537 MVLOGGER BBMZL057I INITIATING FORMAT OF LOG 103294
```

**Figure 24: LOGRPT view**

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) -------
COMMAND ===> SCROLL ===> CSR
CURR WIN ===> 1 ALT WIN ===> 
>W1 =LOGRPT============CXTST2====*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D===1000
- Log Report - Hide Fixed Section
Max Files.. 2 Max Search... 50000
Start Date. CURRENT Start Time... CURRENT
Direction.. REV # of Records. 1000
First Date: 2009/01/12 First Time:.. 16:06:41.284
Last Date: 2009/01/12 Last Time:.. 16:05:49.280
C # Log Log Jobname Log
- - Date------ Time-------- -------- Message ---
2009/01/12 16:05:49.280 MAO161   -JOBNAME  STEPNAME PROCSTEP RC EXCP
2009/01/12 16:05:49.286 MAO161   -MAO161  DFS1929I * DBWP = 49152
2009/01/12 16:05:49.286 MAO161   -MAO161  DFS1929I * DC = 000
2009/01/12 16:05:49.294 I10YIMSG DFS1929I * DESC = 7
2009/01/12 16:05:49.294 MAO161   IEF404I MAO161 - ENDED - TIME=16.05.49
2009/01/12 16:05:49.301 MAO161   -MAO161 ENDED. NAME=Check1stOne
2009/01/12 16:05:49.308 I10YIMSG DFS1929I * DFSDF =
2009/01/12 16:05:49.308 MAO161   $HASP309 INIT 2 INACTIVE ******** C=K1
```

**Figure 25: LOGJRNL view**

```
ddmmmyyyy hh:mm:ss ------ MainView WINDOW INTERFACE (Vv.r.mm) -------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===> 
>W1 =LOGJRNL==========CXTST2====*========20MAR2008==11:15:24====PLEXMGR==D===25
C # Log Log Origin Log
- - Date------ Time-------- -------- Message ---
2009/01/12 16:05:49.280 MAO161   -JOBNAME  STEPNAME PROCSTEP RC EXCP
2009/01/12 16:05:49.286 MAO161   -MAO161  DFS1929I * DBWP = 49152
2009/01/12 16:05:49.286 MAO161   -MAO161  DFS1929I * DC = 000
2009/01/12 16:05:49.294 I10YIMSG DFS1929I * DESC = 7
2009/01/12 16:05:49.294 MAO161   IEF404I MAO161 - ENDED - TIME=16.05.49
2009/01/12 16:05:49.301 MAO161   -MAO161 ENDED. NAME=Check1stOne
2009/01/12 16:05:49.308 I10YIMSG DFS1929I * DFSDF =
2009/01/12 16:05:49.308 MAO161   $HASP309 INIT 2 INACTIVE ******** C=K1
```
Displaying logged messages

For more information about any field in the LOGMSG, LOGRPT, or LOGJRNL view, place the cursor on the field and press the Help key.

LOGMSG, LOGRPT, and LOGJRNL display a screen-size list of messages. You can display the next set of messages by scrolling UP or DOWN. Scrolling DOWN displays more recent messages. Scrolling UP displays older messages. You can also scroll to the right to see additional information about the messages.

To display the complete message text and other detailed information about a specific message, you can:

- Place the cursor on any highlighted field and press Enter
- Type S in the line command field next to the message that you want to display and press Enter

The Log Message Detail dialog is displayed.

**Figure 26: Log Message Detail dialog**

The Log Message Detail dialog is especially useful for reviewing long messages or multiline WTOs. The complete message text is displayed in the Message Text area.
Locating messages by criteria

The LOGMSG, LOGRPT, and LOGJRNL views display large numbers of messages at one time.

Often you are interested in a subset of messages that share certain characteristics or relate to a specific event. Use the LOGPROF command to locate messages by message prefix, job name or number, user ID, or a variety of other criteria. With LOGPROF, you can set up profile criteria that limits the LOGMSG, LOGRPT, and LOGJRNL views to just those messages in which you are interested.

To use the LOGPROF command

1. On the COMMAND line of the LOGMSG, LOGRPT, or LOGJRNL view, type LOGPROF and press Enter.

The Message Log Profile dialog is displayed.

Figure 27: Message Log Profile dialog

<table>
<thead>
<tr>
<th>Message Prefix(s).</th>
<th>Job Name(s). . . .</th>
<th>Userid(s). . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message Level(s) .</td>
<td>Job Number(s). . .</td>
<td>Message Log Profile</td>
</tr>
<tr>
<td></td>
<td></td>
<td>COMMAND  ===&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SCROLL ===&gt; PAGE</td>
</tr>
</tbody>
</table>
|                   |                   | MORE: +

2. Specify profile criteria in one or more of the fields.

The criteria are combined (logically ANDed) to form a profile by which messages are selected for display. For a complete description of the fields in the Message Log Profile dialog, press the Help key.
3 Press the **END** key to save the profile criteria and return to the LOGMSG, LOGRPT, or LOGJRNL view.

The LOGMSG, LOGRPT, or LOGJRNL view contains only those messages that match the profile criteria that you specified. To change the profile criteria or clear the criteria altogether, use the LOGPROF command again.

---
**Note**
Profile criteria are in effect only for as long as the LOGMSG view is displayed. If you display another view and then return to the LOGMSG view, no profile criteria are in effect.

---

**Locating messages by time**

By default, the LOGMSG, LOGRPT, and LOGJRNL views display the most recently recorded messages.

Sometimes you might want to locate messages that were recorded at a specific time on a specific date. For example, while diagnosing a system problem you might want to display messages that were recorded at 10:30 P.M. last night. The LOGTIME command enables you to search for messages by time and date and position the view to the selected start or end time.

**To use the LOGTIME command**

1 On the **COMMAND** line of the LOGMSG, LOGRPT, or LOGJRNL view, type **LOGTIME** and press **Enter**.

The Log Control Profile dialog is displayed.

**Figure 28: Log Control Profile dialog**

<table>
<thead>
<tr>
<th>COMMAND  ===&gt;</th>
<th>Log Control Profile</th>
<th>SCROLL ===&gt;</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Search Criteria:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Direction of Search:</td>
<td>Fwd or Rev</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time:</td>
<td>CURRENT, or hh:mm:ss.ttt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date:</td>
<td>CURRENT, or yyyy/mm/dd</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **END** to exit saving changes
- **CANCEL** to exit without saving changes
- **HELP** to view related help
- **RESET** to reset to initial data

2 Specify the search criteria as follows:

   a In the **Direction of Search** field, specify forward (**F**) or reverse (**R**).

   The default is **F** (forward).
A forward search uses the Time and Date values as the starting point; messages recorded at that time and later are displayed. A reverse search uses the Time and Date values as the ending point; messages recorded at that time and earlier are displayed.

b) In the Time field, specify the starting time (if Direction is forward) or ending time (if Direction is reverse) of the messages to be displayed. The default is CURRENT.

c) In the Date field, specify the starting date (if Direction is forward) or ending date (if Direction is reverse) of the messages to be displayed. The default is CURRENT.

3 Press the END key to save the search criteria and return to the LOGMSG, LOGRPT, or LOGJRNL view.

The LOGMSG, LOGRPT, or LOGJRNL view is positioned to the selected start or end time. You can display additional messages by scrolling UP or DOWN.

Issuing operator commands

You can issue operator commands from the LOGMSG, LOGRPT, or LOGJRNL view or any other view of messages that were logged by MainView Logger.

The command is sent to the operating system, and the result is displayed in the LOGRESP view. The LOGRESP view displays only those messages that use the CART value assigned by the operating system when the command is processed.

Note
Not all command processors use the CART value. To see the output of such commands, use the LOGMSG, LOGRPT, or LOGJRNL view.

To issue an operator command

1 On the COMMAND line of the LOGMSG, LOGRPT, or LOGJRNL view, type an operator command in the following format and press Enter:

/cmd_text

where cmd_text is the complete text of the operator command.
The LOGRESP view is displayed.

Figure 29: LOGRESP view

ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) --------------
COMMAND ===>
SCROLL ===>
PAGE
CURR WIN ===> 1
ALT WIN ===>
>W1 =LOGRESP==MVSB====*========ddmmmyyyy==hh:mm:ss====PLEXMGR==D====5
C # Log Log
- - Time---- Message---
  08:11:01 D J,L
  + 08:11:01 IEEI14I 08.11.01 2005.145 ACTIVITY 371
  + 08:11:01 AA006XX AA006XX STEP01 OWT S AA00MTMC DMTMC DMTOM NS
  + 08:11:01 AA006C5E AA006C5E STEP01 OWT S AA006C5D AA006C5D STEP01 OW
End of Data 214 records in forward direction (BBMZL201I)

Exporting log records

Log records can be extracted for a given time range to create a new log file called an export log file.

Because an export log file spans a limited time range, it is generally much smaller than the log file where the records were originally written. Export log files can be created and sent to BMC for analysis of any problems that might arise with your MainView products. BBSAMP member MVLXPJCL contains sample JCL for exporting log records.

Note

MainView Logger must be active at the time an export log file is created.

Controlling MainView Logger operation

You might need to modify the operation of the MainView Logger logspace while it is running.

For example, you can query the NGL agent to view a list of active logsets. Or you might need to start or stop the NGL agent. You can also use a MODIFY command for the DBC to control the MainView Logger logspace within the address space.
without having to restart the DBC, NGL or MainView logger agents. For more information, see the *BMC Infrastructure Components Administration Guide*.

# Commands that control the MainView Logger logspace

You can use the following commands to control the MainView Logger logspace where the *dbcssid* represents the subsystem identifier (SSID) of the DBC.

<table>
<thead>
<tr>
<th>Function</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>To start the DBC subsystem as a z/OS started task</td>
<td>START <em>dbcssid</em></td>
</tr>
</tbody>
</table>
| To stop the DBC subsystem | Enter one of the following IBM z/OS commands:  
  - *dbcssid* STOP  
  - *group* STOP  
  - MODIFY *dbcssid*, STOP |
| To stop the MainView Logger agent in the DBC | *dbcssid* MVL,STOP  
You can also use the MVS modify command:  
F *dbcssid* MVL,STOP |
| To re-start an MainView Logger agent that has been previously stopped | *dbcssid* MVL,START  
You can also use the MVS modify command:  
F *dbcssid* MVL,START |
| To display the defined logset characteristics to NGL | *dbcssid* NGL,CMD, *piid*,DISPLAY LOGSET *logsetName* |
| To list the logset that is currently in use by the NGL MainView Logger Agent | Use the NGL command:  
*dbcssid* NGL,CMD,*piid*,LIST LOGSET [ *logsetName*]  
You can also use the MVS modify(F) command:  
F *dbcssid* NGL,CMD, *piid*,LIST LOGSET [ *logsetName*]  
The value of the *piid* is the name of the NGL instance and the *logsetName* value is optional.  
If you do not specify a logsetName, all of the logsets are listed. |
| To display insert and retrieval statistics for a logset | *dbcssid* NGL,CMD, *piid*,STATUS LOGSET *logsetName* |
Managing targets and target contexts

Tip
The information in this chapter applies to specific MainView products as described in each topic.

Overview of targets

A target is a system or subsystem that is being monitored by one or more MainView products, such as a system image, IMS or DB2 subsystem, or CICS region.

A target context is a single target and product combination.

When a product is running in full-screen mode, the target ID is displayed in the upper-right corner of the screen in the TGT field. When a product is running in windows mode, the target context is displayed on the window information line. In windows mode, you also can view data from several targets at once and the context that is displayed on the window information line is called an SSI context. See “Managing SSI contexts” on page 183 for more information about defining and displaying SSI contexts.
Target IDs are defined in the BBI-SS PAS in BBPARM member BBIJNT00 for applications that run in full-screen mode. Target contexts are defined with the Plex Manager TGTDEF and TGTDEFD views for applications that run in windows mode.

**Note**

For MainView for CICS, new targets can be dynamically defined by using the Plex Manager Add Target Definition dialog from the TGTDEF view. The target definitions are saved in BBPARM member BBMTXPnn, where nn is a number from 00 to 99. The definitions are appended to an in-memory list of target entries that is produced from member BBIJNT00. Existing targets cannot be modified with the TGTDEF dialog.

---

**Managing targets**

**Tip**

The information in this topic applies to the following products only:

- MainView FOCAL POINT
- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for MQ

Targets are defined in BBPARM member BBIJNT00, as described in the *MainView Customization Reference*. Each target must be associated with the subsystem ID (ssid) of the BBI-SS PAS that monitors it. A single BBI-SS PAS can monitor multiple targets of different types.

All targets (both local and remote) to be accessed from terminal sessions that are connected to a BBI-SS PAS must be defined in the BBIJNT00 member for that BBI-SS PAS. The target and subsystem ID values are used to route user requests to the correct BBI-SS PAS. To simplify maintenance, you might want to maintain a common BBIJNT00 member that includes all targets.
Alternatively, for MainView for CICS, a dialog can be used to define and add targets (see “Managing target contexts” on page 151). Target definitions that are created with this dialog are activated only when the INSTALL command is entered.

When INSTALL is entered or a PAS first starts, the PAS contacts the CAS and is notified about new target definitions. The PAS retrieves a list of defined targets and targets that are active on other PASs from the CAS. It appends this information to the list of targets that was built during startup from BBPARM member BBIJNT00.

**Note**
Existing MainView for CICS targets that are defined in BBIJNT00 cannot be changed with the dialog. If this dialog is used to change targets that are already defined in BBIJNT00, the change is ignored.

When a BBIJNT00 member is changed, the BBI-SS PAS must be restarted for the changes to take effect.

A default target name for your terminal session is set by the TARGET parameter in member BBITSP00 of the BBPROF data set. A matching target name must be defined in BBPARM member BBIJNT00.

## Managing target contexts

This topic explains how to manage target contexts for products when operating in windows mode:

- “Target context definition” on page 151 explains how to define a target context.
- “Managing target context definitions” on page 159 describes how to manage target contexts after they have been defined and recognized by Plex Manager.

**Tip**
The information described in this section applies to all MainView products except MainView FOCAL POINT.

## Target context definition

You can use a target context definition for targets that are critical to your site.

When you define a target context, you specify a unique combination of target and product. Defining a target context allows you to:
Specify products that you expect to be active for a target.

If a product is not active at a target, a signal can be made to MainView AutoOPERATOR to start that product.

Display the status of that target and product combination.

A CAS monitors active products automatically when it initializes. Specifying a target context definition causes the CAS to report the status of a product even if that product is not active.

Define a security profile for that target context.

To define target contexts

1. Use the information in “Planning a target context definition” on page 152 to plan for the necessary target context definitions.

2. Create target definitions by following the instructions in “Defining a target context” on page 154.

3. If necessary, enable target monitoring by following the instructions in “Enabling a target context definition member in the local CAS” on page 157 to define a target definition member to the local CAS definition.

Planning a target context definition

Target context definitions impose certain requirements.

This topic describes issues that must be considered as part of the overall planning of a target context definition. When you plan your target context definitions, consider the following issues:

- Verify that you are authorized to access each system.

- Ensure that cross-system communication is established between the local CAS and each remote CAS that provides data from a target context that is part of the definition. If VTAM communication links are not established between CASs, data cannot be accessed from those target contexts. For information about establishing cross-system communication, see the MainView Customization Reference.

- Determine whether you need to create a target context definition for local or remote products that monitor critical systems.

The CAS reports products that are active. If you want to know about a product that is not active, you can create a target context for that combination of target and product. A target context allows the CAS to report on product availability through the Plex Manager views.
Target context definitions allow you to define a unique security profile for each target even when multiple targets are monitored by the same BBI-SS or z/OS PAS. If you need to create a target context definition in order to apply a specific security resource definition member to a product, determine the two-digit suffix of the security resource member (created through the SERDEF view for the product) by displaying the SERDEFL view in that product.

If a security profile for a target context is not specified, the product security resource definition 00 suffix is used. If that does not exist in the BBSECURE data set, the default security resource definitions that are supplied by BMC are used.

The default security configuration provides security calls for all resources in your product. In a shared DASD environment where all PASs across systems share the same security parameter library, you may need to define a different security resource definition member to one or more products. For example, you might want your production system products to have different security parameters than your test system products. Because CMF MONITOR Online and MainView for z/OS share the same security parameter library on each system image, you can define a default security resource definition member to CMF MONITOR and a different security resource definition member to MainView for z/OS— or vice versa.

If you change the security resource parameter member for a product, you must recycle the address space that supports that product to make the change effective:

— For Plex Manager, the CAS must be recycled.

— For the following products, the z/OS PAS must be recycled:

  — CMF MONITOR
  — MainView for z/OS
  — MainView for UNIX System Services
  — MainView VistaPoint
  — MainView SYSPROG Services

— For the following products, the BBI-SS PAS must be recycled:

  — MainView AutoOPERATOR
  — MainView for CICS
  — MainView for DB2
  — MainView for DBCTL
  — MainView for IMS Online
—MainView for MQ

—For all MainView products (except MainView FOCAL POINT), the product-specific PAS must be recycled.

By default, target context definitions are added to a BBPARM library member with a suffix of 00.

If the default member is used, the instructions in “Enabling a target context definition member in the local CAS” on page 157 can be skipped.

■ In a shared DASD environment, if all CASs use the same target context definition member and share the same parameter library, you need to maintain only one definition member.

For customization or maintenance, you can

— define or change the target context definitions in a target context definition member for one CAS

— enter a single command from the TGTDEF view of all other CASs

— recycle the affected product PASs (necessary only if the security resource parameter has been changed)

**Defining a target context**

A target context definition for each MainView product must be defined in a BBPARM member so you can monitor the corresponding targets across multiple system images.

Later, you may need to add new definitions or maintain existing definitions.

Plex Manager provides a set of views to define target contexts and maintain them after they have been created. The following target context definition views are available in Plex Manager:

■ TGTDEFL lists all existing BBPARM members that contain target context definitions; see “Displaying a list of BBPARM target context definition members” on page 160.

■ TGTDEF lists target context definitions that are contained in a single BBPARM member. Use this view to add, change, generate, update, or delete target context definitions. The changes to the definition can be saved to the BBPARM member.

■ TGTDEFD shows the details of a single target context definition in one BBPARM member. You can perform the same functions to a single definition in TGTDEFD that you can perform to all definitions in TGTDEF.
See “Managing target context definitions” on page 159 for more information about target context definition management tasks.

A target context definition contains:

- One to eight-alphanumeric character target name (special characters are not allowed in the target name)
- Description of the target
- Filter condition(s) that select information about target systems and products defined as a complex filter condition

If you are defining target context definitions for the first time, see “Create a target context definition” on page 155.

Create a target context definition

To define target contexts specifically for monitoring by the CAS or to define to a local product a set of security resource calls defined in a security parameter member, BMC recommends that you follow this procedure on each system in your sysplex or multisystem environment:

1. Start the CASs and PASs across systems for the products that you want to create target context definitions for. See “Starting a CAS” on page 21, “Starting the z/OS PAS” on page 39, and “Starting a BBI-SS PAS” on page 57 for CAS and PAS startup instructions, respectively.

2. Display the MainView Selection Menu on your local system by executing the MainView CLIST and select option P for Plex Manager.

   The EZPLEX menu is displayed.

3. On the COMMAND line, type TGTDEF SUFFIX(00) and press Enter to display the TGTDEF view with the default 00 suffix parameter member.

   Note
   If you do not define the SUFFIX(00) parameter when requesting the TGTDEF view, the following message is displayed:
   BBMYA121I Target definition member nn does not exist in BBPARM

4. On the COMMAND line, type EDIT and press Enter to obtain the edit lock.

5. On the COMMAND line, type GENERATE and press Enter to automatically add a definition for each active product displayed from the PLEX view.
If you need to define additional target contexts that are not defined using the GENERATE command, type ADD on the COMMAND line and press Enter to display the Add New Target Definition dialog, as shown in Figure 30 on page 156.

**Figure 30: Add New Target Definition dialog**

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>Add New Target Definition</th>
<th>SCROLL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Specification:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target Name Target</td>
<td>Name to uniquely identify the target</td>
<td></td>
</tr>
<tr>
<td>Product MainView Product Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System * CAS System Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Server * PAS Server Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description Add your description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security Specification:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID 00 Product Resource Suffix (2 digits)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User ID Default userid, GLOBAL, or blank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name Applid for this Target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>END to exit saving changes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit without saving changes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Add New Target Definition dialog allows you to define a MainView target and product combination (target context). You can use this dialog to:

- Change or add new targets for an z/OS PAS or add new targets for a BBI-SS PAS that is running MainView products in windows mode
  MainView for CICS uses this information to define targets dynamically. This is in addition to targets that are defined by using BBPARM member BBIJNT00.

- Provide security information for local or cross-system products that are sharing a security parameter library

Target information for the Add New Target Definition dialog includes:

- One- to eight-alphanumeric character name of a target to be managed by a z/OS or BBI-SS PAS

- MainView product that is monitoring the specified target

- Name of the local CAS or * if a CAS name is not required to uniquely identify the target

- Name of the PAS server or * if a server name is not required to uniquely identify the target

- Description of the target system:

Security information includes:

- Suffix of the security parameter library member that is used to secure resources for the product
- Default security user ID
- Security application ID for the product

7 Provide the appropriate information for each product.

If you need more information about a field, press the Help key.

8 Enter END to return to the TGTDEF view.

9 Repeat steps 6 through 8 for each MainView product that you want the CAS to specifically monitor. Also, repeat steps 6 through 8 for each local product for which you need to define a specific security resource BBPARM member. (With a shared BBPARM parameter library, you can define a security resource BBPARM member to the target context definition for any product, not just for local products.)

10 On the COMMAND line, type SAVE and press Enter to save the information to the target context definition member in BBPARM.

11 On the COMMAND line, type INSTALL and press Enter to dynamically activate new target context definitions.

12 Perform one of the following actions:

- If each CAS has a unique BBPARM parameter library, repeat steps 2 through 11 on each CAS.

- If all CASs share the same BBPARM parameter library:
  - Repeat steps 2 and 3 on each CAS or open a window to the Plex Manager on each CAS and display the TGTDEF view.
  - Repeat step 11 on each CAS.

**Enabling a target context definition member in the local CAS**

After defining and installing a target context definition member, the local CAS definition must be updated to use the desired member.

The default target context definition member that is assigned to a CAS definition is 00. If you update the 00 member with the target context definitions, you do not need to change the local CAS definition. If you create a new member with a suffix other than 00, the local CAS must be updated.
To enable a specific target context definition in the local CAS

1 Refer to the instructions in “Managing cross-system communication between BBI-SS PASs” on page 116 and perform the following steps.

2 Display the CASDEF view and obtain the edit lock.

3 On the COMMAND line, type CHANGE and press Enter for the local CAS definition.

4 Modify the value in the TgtDef Suffix field of the Change CAS System Definition dialog to be the two-digit value of the target context definition member that you want the local CAS to use.

5 Enter the END command to return to the CASDEF view.

6 On the COMMAND line, type SAVE and press Enter to save the update in the CAS definition member in the BBPARM library.

7 On the COMMAND line, type INStall and press Enter to immediately update the CAS definition in the local CAS, adding the new target context definition member.

The local CAS will now use the new target context definition member that you specified.

In addition, if you changed the Resource Definition Parameter Member, Default Userid, or Security Applid fields for a target context definition in the member, the address space that supports the product must be recycled:

- For Plex Manager, the CAS must be recycled.
- For the following products, the z/OS PAS must be recycled:
  - CMF MONITOR
  - MainView for z/OS
  - MainView for UNIX System Services
  - MainView VistaPoint
  - MainView SYSPROG Services
- For the following products, the BBI-SS PAS must be recycled:
  - MainView AutoOPERATOR
MainView for CICS
MainView for DB2
MainView for DBCTL
MainView for IMS Online
MainView for MQ

For all MainView products (except MainView FOCAL POINT), the product-specific PAS must be recycled.

Managing target context definitions

This topic consists of a series of ordered procedures to modify an existing target context definition in response to changes in the target systems at your site.

Table 16 on page 159 lists common tasks to manage MainView target context definitions. Each of these tasks is described separately in a subsequent topic of this chapter.

Table 16: Target context definition management tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>View or Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>View a list of all target context definition members defined in BBPARM</td>
<td>TGTDEFL view</td>
<td>“Displaying a list of BBPARM target context definition members” on page 160</td>
</tr>
<tr>
<td>Select a target context definition member for display</td>
<td>TGTDEFL view</td>
<td>“Displaying a target context definition member” on page 161</td>
</tr>
<tr>
<td>delete a target context definition member from the BBPARM library</td>
<td>TGTDEFL view</td>
<td>“Deleting a target context definition member” on page 162</td>
</tr>
<tr>
<td>List the target context definitions in a target context definition member</td>
<td>TGTDEF view</td>
<td>“Accessing target context definitions (TGTDEF view)” on page 162</td>
</tr>
<tr>
<td>Add a new target context definition member to the BBPARM library</td>
<td>SUFIX(nn) parameter of the TGTDEF view command</td>
<td>“Adding a new target context definition member” on page 163</td>
</tr>
<tr>
<td>View the details of one target context definition defined in a target definition member</td>
<td>TGTDEFD view</td>
<td>“Viewing target context definition details (TGTDEFD view)” on page 164</td>
</tr>
<tr>
<td>Obtain the edit lock to perform any command except INSTall</td>
<td>EDIT primary command</td>
<td>“Editing target context definitions” on page 165</td>
</tr>
<tr>
<td>Task</td>
<td>View or Command</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Automatically generate target context definitions for all active</td>
<td>GENERATE primary command</td>
<td>“Generating target context definitions” on page 166</td>
</tr>
<tr>
<td>targets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Add a target context definition</td>
<td>ADD primary or A line</td>
<td>“Adding a target context definition” on page 167</td>
</tr>
<tr>
<td>Change an existing target context definition</td>
<td>CHAnge primary or C line</td>
<td>“Changing a target context definition” on page 168</td>
</tr>
<tr>
<td>Copy the contents of one target context definition member into another</td>
<td>COPY primary command</td>
<td>“Copying target context definition members” on page 169</td>
</tr>
<tr>
<td>Cancel edit mode, adds, or changes in progress</td>
<td>CANcel primary command</td>
<td>“Canceling target context definition edits” on page 170</td>
</tr>
<tr>
<td>Delete a target context definition from a BBPARM member</td>
<td>DELete primary or DEL</td>
<td>“Deleting a target context definition” on page 170</td>
</tr>
<tr>
<td>Recover a previously deleted target context definition</td>
<td>UND line command</td>
<td>“Recovering a target context definition” on page 171</td>
</tr>
<tr>
<td>Immediately update the local CAS with new or changed target context</td>
<td>INSTall primary or INS</td>
<td>“Installing changes to a target context definition” on page 172</td>
</tr>
<tr>
<td>definitions</td>
<td>line command</td>
<td></td>
</tr>
<tr>
<td>Permanently save changes to a BBPARM parameter library target context</td>
<td>SAVE primary command</td>
<td>“Saving a target context definition” on page 173</td>
</tr>
<tr>
<td>definition member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deploy (copy) target context definitions to other system</td>
<td>DEPloy primary command</td>
<td>“Deploying target context definitions” on page 174</td>
</tr>
<tr>
<td>Manage and monitor deployment</td>
<td>DPLYMNT view</td>
<td>“Managing and monitoring the deployment process” on page 175</td>
</tr>
</tbody>
</table>

### Displaying a list of BBPARM target context definition members

TGTDEFL displays a list of the existing BBPARM members containing target context definitions that belong to the BBPARM library defined for the local CAS.

Use this view to select and delete members from the parameter library:

- To select a target context definition member for display in the TGTDEF view, see “Displaying a target context definition member” on page 161.

- To delete a target context definition member from the BBPARM library, see “Deleting a target context definition member” on page 162.
To access the TGTDEFL view

1 Perform one of the following actions:

- In Plex Manager, type TGTDEFL on the COMMAND line and press Enter.

- From any MainView product that is running in windows mode, use the following CONtext command to access Plex Manager and display the TGTDEFL view:

  CON * PLEXMGR;TGTDEFL

TGTDEFL appears in a window, as shown in Figure 31 on page 161.

Figure 31: TGTDEFL view

<table>
<thead>
<tr>
<th>ddmmmyyyy hh:mm:ss</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>W1 =TGTDEFL_________MVSB=*ddmmmyyy==hh:mm:ss===PLEXMGR==D==3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMD Member Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--- Suffix ---------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00 All Target Context Definitions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01 Production Target Context Definitions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 SysA Test Target Context Definition</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note

When no members exist in the BBPARM data set (such as the first time you display this view), the following message appears in the Description field of the TGTDEFL view: Empty member--Select to create

For more information about any field on the TGTDEFL view, place the cursor on the field and press the Help key.

See “Managing target contexts” on page 151 for more information about TGTDEFL commands.

Displaying a target context definition member

Enter the Select command from the TGTDEFL view to display a target context definition member.

When the Select command is used, the TGTDEF view, shown in the TGTDEF view figure in “Accessing target context definitions (TGTDEF view)” on page 162, is displayed with the contents of the specified target member.

The following forms of the Select command are available:
Primary command
The primary command is entered on the COMMAND line and requires an nn suffix value to identify the member that you want to display, as follows:

S nn

Line command
The S line command is entered in the line command field of the TGTDEFL view. The action is taken against the member on the line where the command is entered.

Deleting a target context definition member

Enter the DELete command from the TGTDEFL view to delete a target context definition member from the BBPARM parameter library.

When the DELete command is used, the target is removed from the member list shown in TGTDEFL.

The following forms of the DELete command are available:

Primary command
The primary command is entered on the COMMAND line and requires an nn suffix value to identify the member that you want to delete, as follows:

DEL nn

Line command
The DEL line command is entered in the line command field of the TGTDEFL view. The action is taken against the member on the line where the command is entered.

Accessing target context definitions (TGTDEF view)

This procedure describes how to access target context definitions (TGTDEF view)

To access the TGTDEF view

1 In Plex Manager, perform one of the following actions:

- Display the TGTDEFL view, as discussed in “Displaying a list of BBPARM target context definition members” on page 160, and select a BBPARM member for display, as discussed in “Displaying a target context definition member” on page 161.
Type the following command on the **COMMAND** line and press **Enter**:

```
TGTDEF SUFFIX(nn)
```

where \( nn \) is the two-digit suffix of the BBPARM member that you want to display.

From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the TGTDEF view:

```
CON * PLEXMGR;TGTDEF SUFFIX(nn)
```

TGTDEF appears in a window, as shown in **Figure 32 on page 163**.

**Figure 32: TGTDEF view**

<table>
<thead>
<tr>
<th>ddmmmyyyy</th>
<th>hh:mm:ss</th>
<th>MainView WINDOW INTERFACE (Vv.r.mm)</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURR WIN</td>
<td>1</td>
<td>ALT WIN</td>
<td>W1</td>
<td>TGTDEF</td>
<td>MVSB</td>
</tr>
</tbody>
</table>

TGTDEF view showing messages

<table>
<thead>
<tr>
<th>ddmmmyyyy</th>
<th>hh:mm:ss</th>
<th>MainView WINDOW INTERFACE (Vv.r.mm)</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURR WIN</td>
<td>1</td>
<td>ALT WIN</td>
<td>W1</td>
<td>TGTDEF</td>
<td>MVSB</td>
</tr>
</tbody>
</table>

For more information about any field in the TGTDEF view, place the cursor on the field and press the Help key.

See “Managing target contexts” on page 151 for more information about TGTDEF commands.

### Adding a new target context definition member

This procedure describes how to add a new target context definition member.

**To add a new target context definition member to the BBPARM library**

1. Enter the TGTDEF view command with its **SUFFIX(nn)** parameter, as follows:

```
TGTDEF SUFFIX(nn)
```

where \( nn \) is a two-digit suffix value from 00 to 99 that currently is not being used.

The TGTDEF view appears with messages, as shown in **Figure 33 on page 163**.
These messages appear because a new definition member that does not contain any definitions has been added.

To add target context definitions to a new member

1. Enter the EDIT command; see “Editing target context definitions” on page 165.
2. Perform one of the following actions:
   - Enter the ADD command; see “Adding a target context definition” on page 167.
   - Enter the GENERATE command; see “Generating target context definitions” on page 166.
   - Follow the instructions provided in “Create a target context definition” on page 155.

Viewing target context definition details (TGTDEFD view)

TGTDEF does not show all details about all target products, but you can display the TGTDEFD view to see the details of a single target context definition.

To access the TGTDEFD view

1. Display the TGTDEF view, as described in “Accessing target context definitions (TGTDEF view)” on page 162.
2. Place the cursor in the Target Name field for the target definition that you want to see and press Enter.

The TGTDEFD view appears, as shown in Figure 34 on page 164.

Figure 34: TGTDEFD view

```
ddmmmyyyy hh:mm:ss ----- MainView WINDOW INTERFACE (Vv.r.mm) ----------
COMMAND ===>                                               SCROLL ===> PAGE
CURR WIN ===> 1       ALT WIN ==>>
>W1 =TGTDEF==TGTDEFD==MVSB=======*=============(00 BROWSE        )====PLEXMGR==D==1
Target Details:
  Target Name... MVSA  Cas Name.... MVSA  Security Info...
  Alias....... VPASN  Product..... MVVP  Applid........ *NONE*
  Release ...... 1.1   Server...... MVSPASA  Dflt Userid. *NONE*
  IMS type ..... DBDC  Description: System A - MainView VistaPoint
  SerDef Suffix. 00  Update Info....
  UpdSystem..... MVSB  UpdTime..... hh:mm:ss
```
For more information about any field in the TGTDEFD view, place the cursor on the field and press the Help key. See “Managing target contexts” on page 151 for more information about TGTDEFD commands.

**Editing target context definitions**

To add, change, generate, delete, undelete, or save a target definition, you must first obtain the edit lock on a target context definition member in the BBPARM library.

**To obtain the edit lock on a new target definition member**

1. Follow the instructions in “Adding a new target context definition member” on page 163 to add a new BBPARM member.

2. Enter the EDIT command on the **COMMAND** line.

**To obtain the edit lock on an existing target definition member**

1. Display the member in the TGTDEF view (see “Accessing target context definitions (TGTDEF view)” on page 162) or display one target context definition from a member in the TGTDEFD view (see “Viewing target context definition details (TGTDEFD view)” on page 164).

2. Enter the EDIT command on the **COMMAND** line.

When the edit lock is obtained, the edit mode status field on the window information line changes from **(00 BROWSE )** to **(00 EDIT )** to indicate that the edit lock is active.

When you are in edit mode, you can:

- Save target context definitions
- Add target context definitions
- Generate automatic target context definitions
- Change target context definitions
- Delete target context definitions
- Undelete target context definitions

See “Managing target contexts” on page 151 for more information.
Generating target context definitions

The GENERATE command automatically creates a target context definition for each active undefined product that is recognized by the local CAS.

You must be in edit mode before you can enter the GENERATE command; see “Editing target context definitions” on page 165.

When GENERATE is used, a target context definition is created for each product that appears on the PLEX view with an Active status. The GENERATE command is a shortcut command for defining target definitions because you do not have to enter the ADD command and complete the Add New Target Definition dialog for each definition individually. As products become active on the PLEX view, you can enter GENERATE and Plex Manager automatically adds definitions for only the undefined active products.

The GENERATE command is entered on the COMMAND line and is entered in the same way on both the TGTDEF and TGTDEFD views.

When GENERATE is used, the following messages can appear for each active target that is currently defined:

```
BBMYA117W Unable to generate the following target context definitions
-Related:BBMXBQ20W Service point definition exists: system context product server
```

Press Enter to clear these messages.

The TGTDEF view is updated with target context definitions for active products that appear in the PLEX view. The TGTDEFD view does not appear to be updated, but when you return to the TGTDEF view, the new target context definitions are listed.

After entering the GENERATE command, you can perform any of the tasks that are shown in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel any new target context definitions</td>
<td>CANcel primary command</td>
<td>“Canceling target context definition edits” on page 170</td>
</tr>
<tr>
<td>Make new target context definitions active immediately</td>
<td>INSTall primary or INS line command</td>
<td>“Installing changes to a target context definition” on page 172</td>
</tr>
<tr>
<td>Save the new definitions to the BBPARM parameter library member</td>
<td>SAVE primary command</td>
<td>“Saving a target context definition” on page 173</td>
</tr>
</tbody>
</table>
Adding a target context definition

To add a new target context definition, use the ADD command.

You must be in edit mode before you can enter the ADD command; see “Editing target context definitions” on page 165.

When the ADD command is used, the Add New Target Definition dialog, shown in the figure in “Defining a target context” on page 154, is displayed either with * (asterisk) values or with the values of an existing target context definition that you want to use as a template for the new definition. How you enter the ADD command determines the field values that appear.

Note
See “Create a target context definition” on page 155 for information about each field in this dialog.

When the information for a new target context definition is complete, press END from the dialog to return to the TGTDEF or TGTDEFD view. The TGTDEFD view does not appear to be updated, but when you return to the TGTDEF view, the new target context definition is listed.

After you have added a new target context definition you can perform any of the tasks that are shown in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel the new target context definition</td>
<td>CANcel primary command</td>
<td>“Canceling target context definition edits” on page 170</td>
</tr>
<tr>
<td>Make the new target context definition active immediately</td>
<td>INSTall primary or INS line command</td>
<td>“Installing changes to a target context definition” on page 172</td>
</tr>
<tr>
<td>Save the new definition to the BBPARM parameter library member</td>
<td>SAVE primary command</td>
<td>“Saving a target context definition” on page 173</td>
</tr>
</tbody>
</table>

The following forms of the ADD command are available:

- Primary command

  The primary form of the ADD command is entered on the COMMAND line and is entered in the same way on both the TGTDEF and TGTDEFD views.

  When the primary ADD command is used, the Add New Target Definition dialog appears with blank fields.
■ Line command

The A line command is entered in the line command field and is entered in the same way on both the TGTDEF and TGTDEFD views. The action is taken against the resource on the line where the command is entered.

The current definition is assumed to be the definition that you want to use as a template. The Add New Target Definition dialog appears with the values for the current definition. You must cancel from this dialog or change at least one of the following fields:

— Target
— Product
— Server
— System

Changing a target context definition

To change an existing target context definition, use the CHAnge command.

You must be in edit mode before you can enter the CHAnge command; see "Editing target context definitions" on page 165.

When the CHAnge command is used, the Change Target Definition dialog is displayed with the details of the specified target context definition, as shown in Figure 35 on page 168. You can modify any field in this dialog, except the following fields:

■ Target
■ Product
■ Server
■ System

For information about the fields in this dialog, press the Help key.

Figure 35: Change Target Definition dialog

<table>
<thead>
<tr>
<th>COMMAND ====&gt;</th>
<th>Change Target Definition</th>
<th>SCROLL ====&gt;</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Specification:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target name SYSE</td>
<td>Name to uniquely identify the target</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Product PLEXMGR</td>
<td>MainView Product Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System SYSE</td>
<td>CAS System Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Server PLEXMGR</td>
<td>PAS Server Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description Target Manager</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security Specification:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
When changes to the description and security parameters for the target context definition are complete, press END to return to the TGTDEF or TGTDEFD view.

After changing a target context definition you can perform any of the tasks that are shown in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel the changes</td>
<td>CANcel primary command</td>
<td>“Canceling target context definition edits” on page 170</td>
</tr>
<tr>
<td>Make the changes active immediately</td>
<td>INSTall primary or INS line command</td>
<td>“Installing changes to a target context definition” on page 172</td>
</tr>
<tr>
<td>Save the changes to the BBPARM parameter library member</td>
<td>SAVE primary command</td>
<td>“Saving a target context definition” on page 173</td>
</tr>
</tbody>
</table>

The following forms of the CHAnge command are available:

- **Primary command**
  
  The primary form of the CHAnge command is entered on the COMMAND line. This command can be used only from the TGTDEFD view.
  
  Because the TGTDEFD view displays only one target context definition at a time, the current definition is assumed to be the definition that you want to change.

- **Line command**
  
  The C line command is entered in the line command field and is entered in the same way on both the TGTDEF and TGTDEFD views. The action is taken against the resource on the line where the command is entered.

### Copying target context definition members

The COPY command copies the contents of one target context definition member into another member.

You must be in edit mode with the member that you want to copy into before you can enter the COPY command; see “Editing target context definitions” on page 165.

The COPY command is entered on the COMMAND line on both the TGTDEF and TGTDEFD views in the following format:

```
COPY nn
```
In the TGTDEF view, \textit{nn} is the suffix of the member that you want to copy from. In the TGTDEFD view, \textit{nn} is the suffix for the member that you want to copy.

Suffix values for each target context definition member are shown on the TGTDEFL view; see the figure in “Displaying a list of BBPARM target context definition members” on page 160.

After you enter the COPY command on the TGTDEF view, the view is updated with the target context definitions that were contained in the member from which you copied.

After you enter the COPY command on the TGTDEFD view, the view does not appear to be updated, but when you return to the TGTDEF view, the target context definitions in the member that you copied from are listed.

**Canceling target context definition edits**

To exit edit mode, the Add New Target Definition dialog, or the Change Target Definition dialog, use the CANcel command.

You can cancel your changes at any time.

If you enter CANcel when either the Add or Change dialog is displayed, you are returned to the TGTDEF or TGTDEFD view and the edit lock remains active.

If you enter CANcel when either the TGTDEF or TGTDEFD view is displayed, the results are as follows:

- You are returned to browse mode; the edit mode status field on the window information line changes from (00 EDIT ) or (00 EDIT MOD ) to (00 BROWSE ) to indicate that the edit lock is no longer active.

- The BBPARM target context definition parameter member is refreshed from storage with either its contents when the edit lock was obtained or the contents since the last SAVE or END command was entered.

The CANcel command is entered on the \textbf{COMMAND} line of the TGTDEF or TGTDEFD views and the Add and Change Target Definition dialogs.

**Deleting a target context definition**

To delete a target context definition from a target definition member, use the DELete command.

You must be in edit mode before you can enter the DELete command; see “Editing target context definitions” on page 165.
The DELete command is entered in the line command field and is entered in the same way on both the TGTDEF and TGTDEFD views. The action is taken against the current definition where the command is entered.

When the DELete command is used, a target context definition no longer appears in the view. It is deleted from the parameter library when a SAVE or END command (or its corresponding PF key) is entered.

After you have deleted a target context definition, you can perform any of the tasks that are shown in the following table:

<table>
<thead>
<tr>
<th>Task</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel the deletion and recover the target context definition</td>
<td>CANcel primary command</td>
<td>“Canceling target context definition edits” on page 170</td>
</tr>
<tr>
<td>Recover the target context definition</td>
<td>PARm DELETED(*) primary command, then UND line command</td>
<td>“Recovering a target context definition” on page 171</td>
</tr>
<tr>
<td>Dynamically remove the definition immediately</td>
<td>INSTall primary or INS line command</td>
<td>“Installing changes to a target context definition” on page 172</td>
</tr>
<tr>
<td>Permanently remove the definition from the BBPARM member</td>
<td>SAVE primary command (or END command)</td>
<td>“Saving a target context definition” on page 173</td>
</tr>
</tbody>
</table>

**Recovering a target context definition**

The UND line command recovers a target context definition that was marked for deletion.

You must be in edit mode as described in “Editing target context definitions” on page 165. As long as you are in edit mode, you can delete and recover target definitions. However, if you delete a target definition and then save or exit the TGTDEF view, you cannot recover the target that you deleted because it is no longer in the saved target definition member in the BBPARM library.

A target context definition is removed from display in the TGTDEF view when the DELete command is used.

**To redisplay a definition**

1. On the COMMAND line, enter the following command:

   PARM DELETED(*)

2. Enter the UND line command against the target context definition.
The UND line command is entered in the line command field in the same way on both the TGTDEF and TGTDEFD views. The action is taken against the current resource where the command is entered.

## Installing changes to a target context definition

The INSTall command can be entered from the TGTDEF or TGTDEFD view.

Enter the INSTall command to immediately update the local CAS and install changes to one or all target context definitions without waiting until the CAS is recycled or the next system IPL.

The INSTall command dynamically updates the runtime version of the target context definitions in the local CAS, if the local CAS is using the currently displayed BBPARM member. In a shared BBPARM environment, you must display the BBPARM member in TGTDEF on each CAS and enter the INSTall command to update the runtime version for each CAS.

### Note

- If you changed the **Resource Definition Parameter Member**, **Default Userid**, or **Security Applid** fields, you must:
  - use the INSTall command to update the target definitions in the local CAS
  - recycle the PAS
- If you used the Add Target Definition dialog to add a new BBI-SS PAS target for MainView for CICS, conditions are as follows:
  - The INSTall command installs any changes that were made to the target list into target tables used by the PAS.
  - The SAVE command saves a copy of these dynamic target definition(s) in BBPARM member BBMTXP00.

No changes are made to any target entries in BBPARM member BBIJNT00.

INSTall does not save changes to the BBPARM member—a SAVE or END command (or its corresponding PF key) updates the BBPARM member; see “Saving a target context definition” on page 173.

The following forms of the INSTall command are available:
Primary command

The primary form of the INSTall command is entered on the COMMAND line.

When INSTall is entered from the TGTDEF view, all of the target context definitions in the currently displayed BBPARM member are updated in the runtime version of the local CAS.

Because the TGTDEFD view displays one target context definition at a time, only the current definition is updated in the runtime version of the local CAS when INSTall is entered from the TGTDEFD view.

Line command

The INS line command is entered in the line command field and is entered in the same way on both the TGTDEF and TGTDEFD views. The action is taken against the current resource where the command is entered.

Saving a target context definition

Use the SAVE command to save changes or deletions to existing target definitions or save new target context definitions to the BBPARM target definition parameter member.

The SAVE command updates the target context definition member in BBPARM for the local CAS or, in a shared BBPARM environment, for all CASs. The next time the local CAS or a CAS in a shared BBPARM environment is initialized, the newly saved target context definition parameters become active. Use the INSTall command (see “Installing changes to a target context definition” on page 172) instead to activate changes immediately to a target context definition member.

The SAVE command is entered on the COMMAND line. You must be in edit mode before you can enter the SAVE command; see “Editing target context definitions” on page 165.

Note

The END command (or its corresponding PF key) also saves any changes that you made while in edit mode.

When the SAVE command is used, the Save Target Definition dialog appears, as shown in Figure 36 on page 174.

If you are saving changes to an existing target context definition member, this dialog shows the previously defined Description field value.
If you are saving a new target context definition member, this dialog appears with a blank Description field.

**Figure 36: Save a Target Definition dialog**

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>SCROLL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter a Description in the space below</td>
<td></td>
</tr>
<tr>
<td>Description CMF on System B</td>
<td></td>
</tr>
<tr>
<td>END to exit saving changes</td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit without saving changes</td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
</tr>
</tbody>
</table>

**To save the target definition member**

1. Enter a description of up to 30 characters; it will appear in the Description field on the TGTDEFL view.

2. Press END to save the description and any changes to the current member in the BBPARM library.

When you return to either the TGTDEF or TGTDEFD view, the edit mode status field on the window information line has changed from (00 EDIT MOD) to (00 EDIT) to indicate that the edit lock is still active but all modifications have been saved.

**Saving CAS definitions**

To save a new CAS definition as a member of the UBBPARM library, or to save changes to existing CAS definitions in the UBBPARM library, enter the SAVE command.

SAVE can be used at any time in edit mode. The edit mode status field on the window information line changes from (00 EDIT MOD) to (00 EDIT) to indicate that the edit lock is still active and all modifications are saved.

**Note**
The END command (or its corresponding PF key) also saves any changes that you made while in edit mode.

**Deploying target context definitions**

With target context definition deployment, you can deploy one or more target context definitions from one file system to one or more file systems for use on other systems.
You can use MainView Explorer or the MainView windows environment to deploy target context definitions.

**To deploy definitions**

1. Display the TGTDEF view.
2. Select one or more definitions that you want to deploy.
   - **MainView windows environment**
     Enter the Tag line command next to one or more definition names. Use the Tag line command to clear a selection.
   - **MainView Explorer**
     — To select one definition, click a definition name.
     — To select multiple definitions, click the **Enable Selections button**, press and hold the **Ctrl** key, and click the definition names.
3. Start the deployment wizard.
   - In MainView windows, enter the **DEPloy** primary command.
   - In MainView Explorer, right-click a selected definition name and choose **Line action => Deploy Definition**.
4. Complete the deployment wizard.

After completing the deployment wizard, the deployment process starts and the DPLYMNT view is displayed showing the status of the request. For more information about the DPLYMNT view, see “Managing and monitoring the deployment process” on page 175.

**Managing and monitoring the deployment process**

Use the DPLYMNT view to monitor and manage the deployment process. All deployment processes are monitored from the DPLYMNT view, including CAS definition deployment. The DPLYMNT view lists:

- All the deployment requests that have been made from the current system
- The status of the deployment requests

From the DPLYMNT view, you can:
Stop, start, delete, and cancel deployment requests by using the available line commands

Display details about a deployment request by using the Select line command

Display details about the status of a deployment request by hyperlinking from the status

Displaying the status of targets and systems

Tip
The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

Plex Manager manages target applications that are monitored by the MainView family products. Plex Manager provides a series of related views that list target applications and the status of the MainView products that monitor their performance. These views are used to assess your site's current MainView environment and also serve as a starting point to examine individual targets.

Note
Only those products and targets that are defined to a CAS are displayed in Plex Manager views.

Figure 37 on page 177 shows an example of MainView products monitoring their respective applications across two system images. Plex Manager views show the status of MainView products by each application's target context that has been defined for monitoring. Although Plex Manager may be defined through a CAS
running on a local system image, the status of MainView products running on remote images is shown also through these views.

Figure 37: Plex Manager status views of MainView products

Understanding types of Plex Manager status views

There are two main types of Plex Manager status views. The first type can be considered detail views that list all target applications defined to the CAS in the current view context.

Summary views are the second type of Plex Manager status views. In the example shown in the PLEXOVER view figure in “Accessing Plex Manager status views” on page 178, two MainView for CICS systems are running on MVS B. Both systems independently monitor different CICS regions. In this case, the Plex Manager summary views show the combined status of MainView for CICS running on MVS B. Summary views are essential to report data gathered from SSI contexts composed of multiples instances of products operating across multiple system images.
PLEX view

This view shows the status of local and remote MainView products. PLEX lists active and inactive products based upon the individual target contexts of products defined to the local CAS.

The PLEX view figure in “Accessing Plex Manager status views” on page 178 shows an example of the PLEX view.

PLEXOVER view

This view summarizes the status of local and remote MainView products. PLEXOVER lists active and inactive targets by context and the status of the product monitoring the target.

The PLEXOVER view figure in “Accessing Plex Manager status views” on page 178 shows an example of the PLEXOVER view.

PLEXAREA view

This view summarizes the status of local and remote MainView products. PLEXAREA lists active and inactive targets grouped by area and shows the total number of products defined for each area.

PLEXPROD view

This view summarizes the status of local and remote MainView products. PLEXPROD lists active and inactive targets grouped by area and shows the number of instances of a particular product on any connected system.

Accessing Plex Manager status views

Use one of the following methods to access the Plex Manager status views.

If you are not in MainView

1 Start a terminal session by performing one of the following actions:
   - Execute the MainView CLIST from the COMMAND line:
     
     TSO EX 'hilevel.UBBSAMP(MainView)'
Start a VTAM or EXCP MainView Alternate Access terminal session that executes the MainView CLIST.

2 Select option P for Plex Manager.

The EZPLEX menu is the default view that appears first when you access Plex Manager from the MainView Selection Menu.

3 On the EZPLEX menu, position the cursor on one of the following Target Activity fields and press Enter:
   - Sum by Area for the PLEXAREA view
   - Sum by Product for the PLEXPROD view
   - Sum by System/Prod for the PLEXOVER view
   - Not Summarized for the PLEX view

If you are in another MainView product that is running in windows mode

1 Use the following CONtext command:

   CON * PLEXMGR;viewName

If you are currently displaying a Plex Manager view

1 Enter viewName on the COMMAND line.

Figure 38 on page 179 and Figure 39 on page 180 are examples of the PLEX and PLEXOVER status views.
### Displaying the status of targets and systems

**Figure 39: PLEXOVER view**

<table>
<thead>
<tr>
<th>ddmmmyyyy</th>
<th>hh:mm:ss</th>
<th>MainView WINDOW INTERFACE (Vv.r.mm)</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>CURR WIN</th>
<th>ALT WIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>W1 =PLEXOVER==SYSB==*==ddmmmyyyy==hh:mm:ss==PLEXMGR==D==50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Context</th>
<th>Product</th>
<th>Description</th>
<th>Status</th>
<th>Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2P</td>
<td>MVDB2</td>
<td>BBCS TEST DB2 V4</td>
<td>Active</td>
<td>SSA6</td>
</tr>
<tr>
<td>DB2P</td>
<td>MVVP</td>
<td>BBCS TEST DB2 V4</td>
<td>Active</td>
<td>SSA6</td>
</tr>
<tr>
<td>DB2X</td>
<td>MVDB2</td>
<td>BBCS PROD DB2 V4</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>DB2X</td>
<td>MVVP</td>
<td>BBCS PROD DB2 V4</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCICS</td>
<td>MVCICS</td>
<td>EMPRISE TECH CICS V3.3</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCICS</td>
<td>MVVP</td>
<td>EMPRISE TECH CICS V3.3</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCICS1</td>
<td>MVCICS</td>
<td>EMPRISE TECH CICS V3.3</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCICS1</td>
<td>MVVP</td>
<td>EMPRISE TECH CICS V3.3</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCIC4</td>
<td>MVCICS</td>
<td>EMPRISE TECH CICS V4.1</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCCIC4</td>
<td>MVVP</td>
<td>EMPRISE TECH CICS V4.1</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>ETCDLSOT</td>
<td>MVCICS</td>
<td>EMPRISE TECH CICS V3.3</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>GUPCICS</td>
<td>MVCICS</td>
<td>GUPTA CICS V3.30</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>GUPCICS</td>
<td>MVVP</td>
<td>GUPTA CICS V3.30</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>GUPCIC4</td>
<td>MVCICS</td>
<td>GUPTA CICS V4.10</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>GUPCIC4</td>
<td>MVVP</td>
<td>GUPTA CICS V4.10</td>
<td>Active</td>
<td>SSA1</td>
</tr>
<tr>
<td>IMSCTL</td>
<td>MVIMS</td>
<td>IMS</td>
<td>Active</td>
<td>SSA6</td>
</tr>
</tbody>
</table>

In each of these views, the **Context** field shows the name of the target context that is recognized by the CAS. The **Status** field shown to the right of the view indicates whether monitoring is active.

For more information about these views, place the cursor on the view name shown on the window information line and press the Help key. Help is also available for the fields on these views. Place the cursor on a field and press the Help key to get a description of the field.

### Navigating with Plex Manager status views

Plex Manager status views are useful navigation aids when you need to work with several MainView products concurrently on multiple systems.

You can display a Plex Manager status view and repeatedly use its hyperlinks to access targets and contexts that are displayed in the view. By convention, the leftmost data field of a view hyperlinks to other views.

#### To utilize the status views

1. Display a Plex Manager view in the current window.

2. Define an open window as the alternate window in the **ALT WIN** field.

3. Hyperlink from the Plex Manager view to display the target application in the alternate window.
Alternatively, you can use the S or E line commands to select a target shown on a Plex Manager status view.

**Securing contexts**

Access to the views for managing contexts and their definitions can be controlled through security resources.

For more information, see the Plex Manager topics in the *MainView Security Reference Manual*. 
Managing SSI contexts

Tip
The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

A context is a frame of reference for the data that you display in views. A context can be limited to a single target application or be expanded to include all recognized targets providing data to a MainView product. In the case where you are viewing multiple targets, you can set the scope of your views to display selected targets within a context.

Contexts that include more than a single target are called single system image (SSI) contexts. An SSI context functions as a high-level filter that selects the data appearing in your views by the following criteria:

- Target name (such as a CICS region, IMS or DB2 subsystem, or z/OS system image)
- Products attached to a coordinating address space (CAS)
- MainView product
- Product attached to a product address space (PAS)
Figure 40 on page 184 shows CICS, IMS, DB2, and z/OS target applications operating on two system images. MainView products are monitoring their respective targets.

**Figure 40: Views set within SSI and target contexts**

The ALL single system image (SSI) context provides data from all targets on all systems in a multi-system environment. ALL is a default SSI context that is available to MainView products, as shown in item (4) in Figure 40 on page 184. In this example, the ALL context represents the combination of all targets across both system images.

Item (1) in Figure 40 on page 184 is also an example of a target context. A context set to CICSPDA1 within a MainView for CICS view is restricted to data collected from a single CICS region.

In either case, the context is a frame of reference that determines the data collected from MainView target applications that appears in views.

An SSI context must be defined in a series of Plex Manager views. The definition describes the filter conditions that select targets for the views set within the SSI context. After that, other requirements must be met before displaying views set to an SSI context. To support a user-defined SSI context definition, ensure that:

- The correct SSI context definition member is defined to the local CAS
- Your user ID is authorized to access each target
Types of SSI contexts

There are the following general types of SSI contexts:

- **Defined** SSI contexts are either predefined by MainView Infrastructure or defined by your site to represent one or more targets for a given product. The following defined SSI contexts are provided with all MainView products:
  
  - *ALL* provides data from all targets on all systems in a multi-system environment. You can modify this definition.
  
  - *CURRSYS* provides data from all targets running on the local system. You can modify this definition.
  
  - *_ALL* provides data from all targets on all systems in a multi-system environment. You can *not* modify this definition.
  
  - *_CURRSYS* provides data from all targets running on the local system. You can *not* modify this definition.

- **Dynamic** SSI contexts are created by MainView Infrastructure or specific MainView products dynamically to represent natural groupings of targets (such as a sysplex). The following dynamic SSI contexts are provided with all MainView products:
  
  - *SYSPLEX* represents all targets for a given product in the sysplex.
  
  - *SYSTEM* represents all targets for a given product connected to a specific CAS. In this case, the system is identified by the CAS name that was assigned in the CASDEF view. Generally, the CAS name is the same as the z/OS system name, but it might not be if there is more than one CAS running on a system.
  
  - *SYSNAME* represents all targets for a given product on a specific LPAR. In this case, the system is identified by the z/OS system name.

Planning considerations

SSI contexts impose certain requirements.

This topic describes several issues that must be considered as part of the overall planning for an SSI context definition. When you plan for SSI context definitions, consider the following issues:

- You must ensure cross-system communication is available between the local CAS and each CAS that provides data from a target system that belongs to the SSI context.
context. If VTAM communication links are not established between CASs, target data cannot be accessed from those systems.

- If you need to define SSI context definitions across systems, BMC recommends the following actions:
  - Use a consistent naming convention to describe the target products that belong to an SSI context.
  - Ensure that each similarly named SSI context definition contains the same configuration of target systems and products.

Both recommendations are easily applied in a shared DASD environment that permits parameter libraries to be shared across systems. If different parameter libraries are used by CASs operating on different systems, separate copies of the SSI context definition must be maintained. These two recommendations ensure SSI context definitions can be maintained easily if each system requires its own definition.

- Use wildcard characters to declare values set in an SSI context definition. You can avoid potential maintenance by using wildcard characters that generically incorporate a range of targets.

- To avoid updating the local CAS definition to point to a different member, add SSI context definitions to the default parameter member with a suffix of 00.

- If all CASs use the same SSI context definition member and share the same parameter library, you need maintain only one definition member.

**SSI context definition views**

SSI context definitions are created with the MainView Plex Manager services.

Plex Manager provides a set of views to define and maintain context definitions:

- CONDEFL lists the suffixes of BBPARM members that contain existing context definitions.

- CONDEF lists the context definitions contained in a single BBPARM member. Use this view to add, cancel, change, copy, or delete context definitions within one member.

- CONDEFD shows the details of a single context definition belonging to a single BBPARM member. You can perform all the same functions in CONDEFD to a single definition that you can perform in CONDEF for all SSI definitions.
An SSI context definition contains

- one- to eight-alphanumeric character context name
- description of the SSI context
- filter conditions that identify target systems and products belonging to the context

Creating an SSI context definition

**Note**
The predefined SSI context ALL can be used to view data from all systems and products that are recognized by the local CAS.

The predefined SSI context CURRSYS can be used to view data from all target systems running on the local system.

If the ALL or CURRSYS context meets the needs of your site, you do not need to define an SSI context.

BMC recommends that you complete this procedure on each system where you want views to display SSI contexts of data.

**To create an SSI context definition**

1. From the MainView Selection Menu, select option P, Plex Manager.

   The EZPLEX menu is displayed.

2. On the EZPLEX menu under **Administration**, position the cursor on **Context Definition** and press Enter to display the CONDEF view, as shown in **Figure 41** on page 187.

   **Note**
   When you display CONDEF on a system for the first time, only the predefined ALL and CURRSYS context definitions exist in default member 00.

3. On the COMMAND line, type EDIT and press Enter to obtain the edit lock.
4 On the **COMMAND** line, type **ADD** and press **Enter** to display the Add Context Definition dialog, as shown in **Figure 42 on page 188**.

**Figure 42: Add Context Definition dialog**

<table>
<thead>
<tr>
<th>COMMAND ===&gt; Add Context Definition</th>
<th>SCROLL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Identification:</td>
<td></td>
</tr>
<tr>
<td>SSI Context</td>
<td></td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Inclusion Filters: (Target included if ANY are true)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Substitution parms to be used in filters</td>
<td></td>
</tr>
<tr>
<td>%1=TGTNAME %2=TGTSYSTEM %3=TGTPRODUCT</td>
<td></td>
</tr>
<tr>
<td>%4=TGTSEVER %5=TGTDESC %6=CURRSYSTEM</td>
<td></td>
</tr>
<tr>
<td>END to exit saving changes</td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit without saving changes</td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
</tr>
</tbody>
</table>

On the Add Context Definition dialog, you specify one or more filter conditions that select the target systems and MainView products to be included in the SSI context.

5 As shown in **Figure 43 on page 188**, complete the dialog as follows:

a Enter a one- to eight-alphanumeric name for your SSI context.

b Provide a description of up to 40 characters.

c Specify one to eight target systems to be included in the SSI context by using inclusion filters.

When the inclusion filters are used by a view, they are ORed together. Individual filters can include multiple criteria that include the AND operator. Each line is one filter. To help save space in a filter definition, you can use the substitution parameters that are shown on the dialog. The inclusion filters are explained in the online Help.

For a description of the fields on this dialog, press the Help key.

**Figure 43: Example of completed Add Context Definition dialog**

<table>
<thead>
<tr>
<th>COMMAND ===&gt; Add Context Definition</th>
<th>SCROLL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Identification:</td>
<td></td>
</tr>
<tr>
<td>SSI Context CMFABC</td>
<td></td>
</tr>
<tr>
<td>Description CMF ON MVSA, MVSB, MVSC</td>
<td></td>
</tr>
<tr>
<td>Inclusion Filters: (Target included if ANY are true)</td>
<td></td>
</tr>
<tr>
<td>1 (TGTPRODUCT IN (CMF,PLEXMGR)) AND (%2 in (MVSA,MVSB,MVSC))</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Substitution parms to be used in filters
%1=TGTNAME %2=TGTSYSTEM %3=TGTPRODUCT
%4=TGTSEVER %5=TGTDESC %6=CURRSYSTEM

END to exit saving changes
CANCEL to exit without saving changes
HELP to view related help
RESET to reset to initial data

6 Press the END key to return to the CONDEF view.

7 On the COMMAND line, type SAVE and press Enter to save the information.

8 On the COMMAND line, type INStall and press Enter to dynamically activate the new SSI context definition.

9 Proceed as follows:

- If all CASs share the same BBPARM parameter library, repeat steps 2, 3, and 8 for each CAS.
- If each CAS has a unique BBPARM parameter library, repeat steps 1 through 8 for each CAS.

10 (optional) If you want the local CAS to use a new SSI context definition that you just installed, take the following steps:

---

**Note**

The name of the default BBPARM member containing SSI context definitions has a suffix of 00. If you updated this member, you do not need to perform the following procedure.

---

a Display the CASDEF view and obtain the edit lock.

b On the COMMAND line, type CHAnge and press Enter.

c Modify the value in the ConDef Suffix field of the Change CAS System Definition dialog to be the two-digit value of the SSI context definition member that you want the local CAS to use.

d Enter the END command to return to the CASDEF view.

e On the COMMAND line, type SAVE and press Enter to save the update in the CAS definition member in the BBPARM library.

f On the COMMAND line, type INStall and press Enter to immediately update the local CAS with the new SSI context definition member.
The local CAS will now use the new SSI context definition member that you specified.

Managing SSI context definitions

The following table lists common tasks to manage SSI context definitions.

Table 17: SSI context definition management tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>View or Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>View a list of all SSI context definition members defined in BBPARM</td>
<td>CONDEFL view</td>
<td>“Managing SSI context definition members (CONDEFL view)” on page 191</td>
</tr>
<tr>
<td>Display the contents of an SSI context definition member</td>
<td>CONDEFL view</td>
<td>“Displaying an SSI context definition member” on page 192</td>
</tr>
<tr>
<td>Delete an SSI context definition member from the BBPARM library</td>
<td>CONDEFL view</td>
<td>“Deleting an SSI context definition member” on page 192</td>
</tr>
<tr>
<td>View the contents of products and targets in an SSI context definition member</td>
<td>CONDEF view</td>
<td>“Viewing products and targets in a definition member (CONDEF view)” on page 193</td>
</tr>
<tr>
<td>Add a new SSI context definition member to the BBPARM library</td>
<td>SUFFIX(nn) parameter of the CONDEF view command</td>
<td>“Adding a new SSI context definition member” on page 193</td>
</tr>
<tr>
<td>View the details of one product monitoring a target defined in an SSI context</td>
<td>CONDEFD view</td>
<td>“Viewing SSI context definition details (CONDEFD view)” on page 194</td>
</tr>
<tr>
<td>Obtain the edit lock to enable you to modify the definitions</td>
<td>EDIT primary command</td>
<td>“Editing an SSI context definition” on page 195</td>
</tr>
<tr>
<td>Copy the contents of one SSI context definition to another</td>
<td>COPY primary command</td>
<td>“Copying an SSI context definition member” on page 195</td>
</tr>
<tr>
<td>Add an SSI context to an SSI context definition member</td>
<td>ADD primary command or A line command</td>
<td>“Adding an SSI context definition” on page 196</td>
</tr>
<tr>
<td>Change an existing SSI context definition</td>
<td>CHAnge primary command or C line command</td>
<td>“Changing an existing SSI context definition” on page 197</td>
</tr>
<tr>
<td>Cancel edit mode and any modifications that were not saved</td>
<td>CANcel primary command</td>
<td>“Canceling modifications to an SSI context definition” on page 198</td>
</tr>
<tr>
<td>Delete an SSI context definition from another SSI context definition member</td>
<td>DELete primary command or DEL line command</td>
<td>“Deleting an SSI context definition” on page 199</td>
</tr>
<tr>
<td>Immediately update the local CAS with new or changed SSI context definitions</td>
<td>INSTall primary command</td>
<td>“Installing changes to SSI context definitions” on page 200</td>
</tr>
</tbody>
</table>
### Managing SSI context definition members (CONDEFL view)

Managing SSI context definitions can be performed using the CONDEFL view. Here are the functions you can perform:

- **Display a list of all SSI context definition members that are defined in BBPARM for the local CAS.**
- **Select an SSI context definition member to display** (see “Displaying an SSI context definition member” on page 192).
- **Delete an SSI context definition member from the BBPARM library** (see “Deleting an SSI context definition member” on page 192).

#### To access the CONDEFL view

1. Perform one of the following actions:
   - In Plex Manager, type **CONDEFL** on the **COMMAND** line and press Enter.
   - From any MainView product that is running in windows mode, use the following CONtext command to access Plex Manager and display the CONDEFL view:
     ```plaintext
     CON * PLEXMGR;CONDEFL
     ```

The CONDEFL view appears, as shown in Figure 44 on page 191.

#### Figure 44: CONDEFL view

```
ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) -----------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 ALT WIN ===>  
WI =CONDEFL ----------- SYSC =**********ddmmmyyyy hh:mm:ss =PLEXMGR =D =**2
CMD Member Description  
--- Suffix -----------  
00 Production SSI Contexts  
01 Test New Context
```
Displaying an SSI context definition member

To display the contents of an SSI context definition member, use the Select command from the CONDEFL view.

After issuing the Select command, the CONDEF view, shown in “Viewing products and targets in a definition member (CONDEF view)” on page 193, is displayed with the contents of the specified SSI context member.

The following forms of the Select command are available:

- Primary command
  - The Select primary command is entered on the COMMAND line and requires an nn suffix value to identify the member that you want to display:
    - \texttt{S nn}

- Line command
  - The S line command is entered in the line command field of the CONDEFL view. The action is taken against the member on the line where the command is entered.

Deleting an SSI context definition member

To delete an SSI context definition member from the BBPARM parameter library, use the DELete command from the CONDEFL view.

After issuing the DELete command, the SSI context is removed from the member list in CONDEFL.

The following forms of the DELete command are available:

- Primary command
  - The DELete primary command is entered on the COMMAND line and requires an \textit{nn} suffix value to identify the member that you want to delete:
    - \texttt{DEL \textit{nn}}

- Line command
  - The DEL line command is entered in the line command field of the CONDEFL view. The action is taken against the member on the line where the command is entered.
Viewing products and targets in a definition member (CONDEF view)

To access the CONDEF view

1. Perform one of the following actions:
   - In Plex Manager, type **CONDEF** on the **COMMAND** line and press **Enter**.
   - From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the CONDEF view:
     ```
     CON * PLEXMGR;CONDEF
     ```

Figure 45 on page 193 shows an example of the CONDEF view.

Adding a new SSI context definition member

To add a new SSI context definition member to the BBPARM library, enter the CONDEF view command with its **SUFFIX(nn)** parameter as follows:

**CONDEF SUFFIX(nn)**

where **nn** is a unique two-digit suffix value from 00 to 99 that is not currently used.
The CONDEF view appears in the window as shown in Figure 46 on page 194. This example is a result of issuing the command CONDEF SUFFIX(03).

**Figure 46: CONDEF view after creating a new SSI definition member**

```
CONDEF  SYSC  (03 BROWSE)  PLEXMGR
BBMYA548I SSI Context Definition Member 03 does not exist; Predefined context ALL will be used
```

This message indicates that the new member is empty except for the predefined ALL context.

**To add SSI context definitions to the new member**

1. Enter the EDIT command; see “Editing an SSI context definition” on page 195.

2. Perform one of the following actions:
   - Enter the ADD command; see “Adding an SSI context definition” on page 196.
   - Enter the COPY command; see “Copying an SSI context definition member” on page 195.
   - Follow the instructions provided in “Creating an SSI context definition” on page 187.

**Viewing SSI context definition details (CONDEFD view)**

The full width of a CONDEF view cannot be shown within a single window.

You can enter a series of Right commands to shift right and scroll through the filter fields. Alternatively, you can display the CONDEFD view and see all the filters listed in a detail view.

**To access the CONDEFD view**

1. Display the CONDEF view.

2. Place the cursor on the name of the SSI context definition that you want to see details about and press Enter.

The CONDEFD view appears as shown in Figure 47 on page 194.
Editing an SSI context definition

To add, change, copy, delete, or save a target definition, you must first obtain the edit lock on the SSI context definition member.

The edit lock enables you to issue other commands that are available with the CONDEF or CONDEFD views.

When the edit lock is obtained, the edit mode status field on the window information line changes from (00 BROWSE) to (00 EDIT) to indicate the edit lock is active.

After you are in edit mode, you can:

- Save changes in an SSI context definition to the BBPARM member
- Add other SSI context definitions to the SSI context definition member that is being edited
- Delete SSI context definitions
- Copy the contents of existing SSI context definition members into the current member
- Change SSI context definitions

Copying an SSI context definition member

To copy the contents of one SSI context definition member into another, use the COPY command.
You must be in edit mode with the member that you want to copy into before you can enter the COPY command; see “Editing an SSI context definition” on page 195.

The COPY command is entered on the COMMAND line on both the CONDEF and CONDEFD views in the following format:

COPY nn

In the CONDEF view, nn is the suffix of the member that you want to copy from. In the CONDEFD view, nn is the suffix of the member that you want to copy.

Suffix values for each SSI context member are shown on the CONDEFL view; see “Managing SSI context definition members (CONDEFL view)” on page 191.

After you enter the COPY command on the CONDEF view, the view is updated with the SSI context names that were contained in the member from which you copied.

After you enter the COPY command on the CONDEFD view, the view does not appear to be updated, but when you return to the CONDEF view, the SSI context names in the member you copied from are listed.

**Adding an SSI context definition**

To add a new SSI context definition, use the ADD command.

You must be in edit mode before you can enter the ADD command; see “Editing an SSI context definition” on page 195.

When the ADD command is issued, the Add SSI Context Definition dialog shown in “Creating an SSI context definition” on page 187 is displayed, either with blank fields or with the values of an existing SSI context definition that you want to use as a template for a new definition. How you enter the ADD command determines the field values that are displayed.

See “Creating an SSI context definition” on page 187 for information about each field in this dialog.

After defining the name, description, and filter conditions of a new SSI context definition, press the END key. Save the definition to the BBPARM parameter library member by issuing the SAVE command; see “Saving an SSI context definition” on page 200. If you want to make the new SSI context definition active immediately, enter the INSTall command; see “Installing changes to SSI context definitions” on page 200.

To cancel a new SSI context definition while you are defining it, enter the CANcel command; see “Canceling modifications to an SSI context definition” on page 198.
The following forms of the ADD command are available:

- **Primary command**
  
  The ADD primary command is entered on the **COMMAND** line and is entered in the same way on both the CONDEF and CONDEFD views.
  
  When the ADD command is issued from the CONDEF view, the Add Context Definition dialog appears with blank fields.
  
  The CONDEFD view displays only one SSI context definition at a time. The current definition is assumed to be the definition that you want to use as a template for a new definition. The Add Context Definition dialog appears with the values for the current definition. You must at least change the **SSI Context** field or enter the CANcel command to exit from this dialog.

- **Line command**
  
  The A line command is entered in the line command field in the same way on both the CONDEF and CONDEFD views. The action is taken against the resource where the command is entered.
  
  The displayed definition is assumed to be the template that you want to use for a new definition. The Add Context Definition dialog appears with the values for the current definition. You must at least change the **SSI Context** field or enter the CANcel command to exit from this dialog.

### Changing an existing SSI context definition

To change the values of an existing SSI context definition, use the CHAnge command.

You must be in edit mode before you can enter the CHAnge command; see “Editing an SSI context definition” on page 195.

After the CHAnge command is issued, the Change Context Definition dialog is displayed, as shown in Figure 48 on page 197. It provides details about the specified SSI context definition. You can modify any field in this dialog, except the **SSI Context** field. See “Creating an SSI context definition” on page 187 for information about each field in this dialog.

### Figure 48: Change Context Definition dialog

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>Change Context Definition</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Identification:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSI Context CMFABC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description CMF PRODUCT ON MVSA, MVS8, SYSC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inclusion Filters: (Target included if ANY are true)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 (TGTPRODUCT IN (CMF,PLEXMGR)) AND (%2 in (MVSA,MVS8,SYSC))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
After changing the description, filter conditions, or both for the SSI context definition, press END from the dialog to save the definition to the BBPARM parameter library member. To make the changes active immediately, enter the INSTAll command; see “Installing changes to SSI context definitions” on page 200.

To cancel your changes, enter the CANcel command; see “Canceling modifications to an SSI context definition” on page 198.

The following forms of the CHAnge command are available:

- **Primary command**
  
  The CHAnge primary command is entered on the COMMAND line. This command is entered differently, depending on whether the CONDEF or CONDEFD view is displayed:

  - When the CONDEF view is displayed, enter the CHAnge command as follows:
    
    \[ \text{CHA ssiname} \]

    where ssiname is the name of the SSI context definition that you want to change, as shown in the SSI Context field of the view.

  - When the CONDEFD view is displayed, enter the CHAnge command as follows:
    
    \[ \text{CHA} \]

    The CONDEFD view displays only one SSI context definition at a time. The displayed definition is assumed to be the definition that you want to change.

- **Line command**
  
  The C line command is entered in the line command field of both the CONDEF and CONDEFD views. The action is taken against the resource where the command is entered.

**Canceling modifications to an SSI context definition**

To cancel changes made to an SSI context during an edit lock, use the CANcel command.
If you enter CANcel when either the Add or Change dialog is displayed, you are returned to the CONDEF or CONDEFD view and the edit lock remains active.

If you enter CANcel when either the CONDEF or CONDEFD view is displayed, the results are as follows:

- You are returned to browse mode; the edit mode status field on the window information line changes from (00 EDIT) or (00 EDIT MOD) to (00 BROWSE) to indicate that the edit lock is no longer active.
- The BBPARMS SSI context definition member is refreshed from storage with the contents of the member since the last SAVE command was issued.

The CANcel command is entered on the COMMAND line of the CONDEF and CONDEFD views and the Add and Change Context Definition dialogs.

## Deleting an SSI context definition

To delete a specific SSI context definition from a BBPARM parameter library, use the DELete command.

You must be in edit mode before you can enter the DELete command; see “Editing an SSI context definition” on page 195.

When the DELete command is issued, an SSI context definition is marked for deletion and is removed from the Plex Manager views. However, the definition is not deleted from the BBPARM SSI context definition member until the SAVE command is issued; see “Saving an SSI context definition” on page 200.

The following forms of the DELete command are available:

- Primary command
  
The DELete primary command is entered on the COMMAND line. This command is entered differently, depending on whether the CONDEF or CONDEFD view is displayed:

  — When the CONDEF view is displayed, enter the DELete command as follows:

    \[
    \text{DEL ssiname}
    \]

    where \textit{ssiname} is the name of the SSI context definition that you want to delete, as shown in the SSI Context field of the view.
— When the CONDEFD view is displayed, enter the DELete command as follows:

```
DEL
```

The CONDEFD view displays only one SSI context definition at a time. The displayed definition is assumed to be the definition that you want to delete.

### Line command

The DEL line command is entered in the line command field and is entered in the same way on both the CONDEF and CONDEFD views. The action is taken against the current definition where the command is entered.

### Installing changes to SSI context definitions

The INSTall command immediately updates the local CAS with changes to SSI context definitions.

INSTall dynamically updates the runtime version of the SSI context definition in the CAS without waiting until the CAS is recycled or the next system IPL.

INSTall does not save changes to the BBPARM member. Use the SAVE command to update the BBPARM member; see “Saving an SSI context definition” on page 200.

After you use the INSTall command, the CONDEFD view does not change to reflect your update. To see the status of your SSI context definitions, display the CONACTZ view.

### Saving an SSI context definition

To save changes or deletions to SSI context definitions in the BBPARM definition member, use the SAVE command.

The SAVE command is entered on the COMMAND line. You must be in edit mode before you can enter the SAVE command; see “Editing an SSI context definition” on page 195.

When the SAVE command is issued, the Save an SSI Context Definition dialog appears, as shown in Figure 49 on page 201:

- If you are saving changes to an existing context definition, this dialog shows the previously defined Description field value.
If you are saving a new context definition, this dialog appears with a blank Description field.

**Figure 49: Saving an SSI context definition member**

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>Save an SSI Context Definition</th>
<th>SCRDL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter a Description in the space below</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Description CMF PRODUCT ON MVSA, MVS8, SYSC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>END to exit saving changes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit without saving changes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**To save the SSI context definition member**

1. Enter a description of up to 30 characters; it will appear in the Description field on the CONDEFL view.

2. Press END to update or define the description and save any changes to the current member in the BBPARM library.

When you return to either the CONDEF or CONDEFD view, the edit mode status field on the window information line changes from (00 EDIT MOD) to (00 EDIT) to indicate the edit lock is still active but all modifications have been saved.

**Deploying SSI context definitions**

With SSI context definition deployment, you can deploy one or more SSI context definitions from one file system to one or more file systems for use on other systems.

You can use MainView Explorer or the MainView windows environment to deploy SSI context definitions.

**To deploy definitions**

1. Display the CONDEF view.

2. Select one or more definitions that you want to deploy.

   - MainView windows environment
     
   Enter the Tag line command next to one or more definition names. Use the Tag line command to clear a selection.

   - MainView Explorer

     —To select one definition, click a definition name.
To select multiple definitions, click the Enable Selections button, press and hold the Ctrl key, and click the definition names.

3 Start the deployment wizard.

- In MainView windows, enter the DEPloy primary command.
- In MainView Explorer, right-click a selected definition name and choose Line action => Deploy Definition.

4 Complete the deployment wizard.

After completing the deployment wizard, the deployment process starts and the DPLYMNT view is displayed showing the status of the request. For more information about the DPLYMNT view, see “Managing and monitoring the deployment process” on page 202.

Managing and monitoring the deployment process

Use the DPLYMNT view to monitor and manage the deployment process. All deployment processes are monitored from the DPLYMNT view, including SSI context definition deployment. The DPLYMNT view lists:

- All the deployment requests that have been made from the current system
- The status of the deployment requests

From the DPLYMNT view, you can:

- Stop, start, delete, and cancel deployment requests by using the available line commands
- Display details about a deployment request by using the Select line command
- Display details about the status of a deployment request by hyperlinking from the status

Displaying the status of SSI contexts

The Plex Manager CONACT, CONACTZ, and CONACTD views show the current monitoring status of targets within single system image (SSI) contexts that are defined to the local CAS.

An SSI context enables users to view multiple targets that are running on several system images in a single view and work with the information as if it came from a
single system. These views show which MainView products are monitoring their target applications for a specific SSI context. This topic describes how to use these views to assess the current monitoring status of targets within an SSI context.

Table 18 on page 203 lists the Plex Manager views that display the monitoring status of SSI contexts. Each view shows the monitoring status with a different level of detail. The views have hyperlinks that allow you to shift between them. The remainder of this topic describes each view in detail.

### Table 18: SSI context activity tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>View</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display a list of all available SSI contexts and the summary status of each MainView product defined to an SSI context</td>
<td>CONACTZ view</td>
<td>“Viewing context and product availability (CONACTZ view)” on page 203</td>
</tr>
<tr>
<td>View the status of each MainView product that is monitoring individual targets within an SSI context</td>
<td>CONACT view</td>
<td>“Viewing target status in an SSI context (CONACT view)” on page 205</td>
</tr>
<tr>
<td>Display the status of a single MainView product that is monitoring a specific application target within an SSI context</td>
<td>CONACTD view</td>
<td>“Viewing an individual target status (CONACTD view)” on page 206</td>
</tr>
</tbody>
</table>

### Viewing context and product availability (CONACTZ view)

The CONACTZ view displays a list of active SSI contexts that are recognized by the local CAS.

It gives a summary status of the MainView products that are monitoring targets within each SSI context. CONACTZ displays the number of targets that are defined to the SSI context for each MainView product and the count of targets that are currently monitored.
Figure 50 on page 204 shows an example of an ALL SSI context that incorporates targets running on two system images. MainView products are running on both images to monitor their respective targets.

**Figure 50: CONACTZ view within the ALL context**

In this example, CONACTZ summarizes monitoring of the targets within the context across both system images for each MainView product. Use the following methods to access CONACTZ:

- In Plex Manager, type **CONACTZ** on the COMMAND line and press Enter.
- From any MainView product that is running in windows mode, use the following CONtext command:

  \[\text{CON} * \text{PLEXMGR;CONACTZ}\]

Figure 51 on page 204 shows a CONACTZ view that summarizes MainView monitoring within the ALL SSI context of Figure 50 on page 204.

**Figure 51: CONACTZ view**

```
ddmmmyyyy  hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) ---------------
COMMAND  ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===> >W1 -CONACTZ-------------SYSB-------------*------------ddmmmyyyy=hh:mm:ss----PLEXMGR==D==7
```

In this example, CONACTZ summarizes monitoring of the targets within the context across both system images for each MainView product. Use the following methods to access CONACTZ:

- In Plex Manager, type **CONACTZ** on the COMMAND line and press Enter.
- From any MainView product that is running in windows mode, use the following CONtext command:

  \[\text{CON} * \text{PLEXMGR;CONACTZ}\]

Figure 51 on page 204 shows a CONACTZ view that summarizes MainView monitoring within the ALL SSI context of Figure 50 on page 204.
For information about any field in the CONACTZ view, place the cursor on the field and press the Help key.

CONACTZ provides a hyperlink to the CONACT view. Place the cursor on the **SSI Context** field of the MainView product that has targets defined within the SSI context and press **Enter**.

**Viewing target status in an SSI context (CONACT view)**

CONACT shows the status of targets monitored by MainView products within an SSI context.

CONACT expands the summarized counts of CONACTZ to show the individual monitoring status of each target.

Alternatively, CONACT also can be displayed by the following methods:

- In Plex Manager, type **CONACT** on the **COMMAND** line and press **Enter**.

- From any MainView product that is running in windows mode, use the following **CONtext** command:

  **CON * PLEXMGR;CONACT**

*Figure 52 on page 205* shows the CONACT view of the CICS targets that are defined to the ALL context in “Viewing context and product availability (CONACTZ view)” on page 203. The CICS targets summarized in the previous CONACTZ view are expanded to show the monitoring status of individual regions. The CICSTEST region is inactive. All remaining regions are active, which agrees with the summary status shown in the previous CONACTZ view.

*Figure 52: CONACT view*
For information about the status of a product monitoring a target that is part of an SSI context, place the cursor on the **Status** field and press the Help key.

CONACT provides a hyperlink to the CONACTD view. Place the cursor on the **SSI Context** field of the MainView product that has targets defined within the SSI context and press **Enter**.

### Viewing an individual target status (CONACTD view)

CONACTD is a detail view.

It shows the monitoring status of an individual application target within an SSI context.
Figure 53 on page 207 shows an example of a single target selected within an SSI context. CONACTD provides more information about monitoring of this specific CICS region within the ALL context.

**Figure 53: CONACTD view of an individual target within an SSI context**

To display a CONACTD view, invoke a hyperlink from the CONACT view by placing the cursor on a product in the **SSI Context** field and pressing **Enter**.

Figure 54 on page 207 shows an example of a CONACTD view that provides more information about monitoring the CICS region shown in Figure 53 on page 207.

**Figure 54: CONACTD view**

For information about any field in the CONACTD view, place the cursor on the field and press the Help PF key.
Securing contexts

The information in this topic applies to all MainView products (except MainView FOCAL POINT).

Access to the views for managing contexts and their definitions can be controlled through security resources. For more information, see the Plex Manager topics in the MainView Security Reference Manual.
Managing historical data sets

**Tip**
The information in this chapter applies to the following products only:
- CMF MONITOR
- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for IP
- MainView for Linux - Servers
- MainView for UNIX System Services
- MainView for VTAM
- MainView for WebSphere Application Server
- MainView for MQ
- MainView for z/OS
- MainView Storage Resource Manager (SRM)
- MainView SYSPROG Services
- MainView Transaction Analyzer
- MainView VistaPoint

This chapter provides information about the MainView historical data sets. You can view historical data by using the TIME command.

**Understanding the historical database**

This section provides information about the historical database and how it operates in the MainView environment so you can better control and manage your data.
What is the historical database?

The historical database is a group of up to 100 VSAM data sets that store the data collected by each PAS.

A unique historical database exists for each PAS. As data is gathered by a PAS’s data collectors, the historical recorder writes data to the PAS’s historical database. One set of records is written by the historical recorder at the end of each interval.

*Note*

With MainView for z/OS version 2.7.00 or later, the PAS can optionally have a second historical recorder and database that are used to record long-term history. For more information about long-term data collection, see the MainView for z/OS Customization Guide.

The historical database consists of two logical components:

- Up to 100 data sets (VSAM clusters), to house data from the recent past and, optionally, to serve as a receptacle for data that has been loaded from archived data.
  
  Each data set is either
  
  — allocated to the PAS by a HISTDSnn DD statement in the PAS started task procedure
  
  — dynamically allocated with the DSLIST view

  *Note*

  For long-term history data sets, the DD statement is HST1DSnn. To access the DSLIST view for long-term history, first use the DSLISTZ view to display information about both historical databases, and then hyperlink to the appropriate DSLIST view.

- An optional archive, consisting of a tape library or offline data sets, to store data that is too outdated or voluminous to maintain online

MainView provides the tools to allocate and maintain historical data sets.

**Historical data in a multisystem, multiproduct environment**

If you have multiple PASs running on multiple systems or you have multiple PASs running on the same system, keep the following issues in mind:
Each PAS on each system requires a unique historical database. Use the system's SMF ID as the second qualifier and the PAS ID as the third qualifier when allocating the VSAM data sets to indicate the system and PAS to which each historical database belongs; for example, hilevel.smfid.pasid.HISTDS01. This distinguishes one system's historical data sets from another's and ensures the integrity of each system's data.

MainView products that share the same PAS on the same system also share the same historical data sets.

**When is historical data recorded?**

If you have allocated historical data sets, historical reporting begins as soon as the PAS is initialized.

For CMF MONITOR and MainView for z/OS, data is written into the historical data sets at the same rate that performance data is recorded by the CMF MONITOR Extractor. That is, if the Extractor interval is set to 30 minutes, data is written to the historical database every 30 minutes as well. (You specify the Extractor rate on the REPORT statement in the CMFCPMxx or CMFIPMxx control member.)

For CMF MONITOR and MainView for z/OS, if IPM recording becomes active for the Extractor, the historical recorder uses the IPM rate, not the CPM rate. This change can result in interval rates of 5- or 10-minute increments and can impact data usage when viewing multiple intervals of historical data in a view.

---

**Note**

If long-term data collection is activated in MainView for z/OS, data is written to the second historical database at a rate that is some multiple of the regular interval described here. For information about how the long-term interval is specified, see the *MainView for z/OS Customization Guide*.

If short-term data collection is activated in MainView for z/OS version 3.0.00 or later and CMF MONITOR version 5.8.00 or later, data is written to another historical database at a rate of once every 15, 30, or 60 seconds.

For products that execute in the BBI-SS PAS (MainView AutoOPERATOR, MainView for CICS, MainView for DB2, MainView for DBCTL, MainView for IMS Online, and MainView for MQ), data is recorded at the interval that is specified with the IRRI parameter in BBPARM member BBIISP00. If the IRRI parameter is not specified in BBIISP00, the default length of the recording interval is 15 minutes.

Intervals are usually synchronized on the hour. For example, if an interval is defined as 15 minutes and extraction begins at 7:49 A.M., the first interval lasts 11 minutes until the hour of 8:00 A.M. is reached. From then on, the intervals are 8:00 to 8:15, 8:15 to 8:30, and so on.
Where is historical data recorded?

Initially, historical data is written to the end of the historical file that contains the most recent data.

You can control subsequent file selection by issuing commands on the DSLIST view; see “Managing the historical database” on page 215.

Initial file selection

When the historical recorder initializes, the first thing it does is select a historical data set to write to.

Under most circumstances, the historical recorder selects the historical data set it was writing to last when the PAS stopped. This data set contains the most recent historical data.

For example, if the data set that is allocated on the HISTDS03 DD statement in the PAS procedure was the last data set written to last night, the historical recorder again writes to that same data set named on the HISTDS03 DD statement the next morning.

If you make the data set named on the HISTDS03 DD statement ineligible for recording by issuing the O line command on the DSLIST view, the historical recorder searches for an empty data set to write to.

If an allocated, empty data set does not exist on any of the HISTDSnn or HST1DSnn DD statements, the recorder selects the data set containing the oldest data and overwrites it.

Subsequent file selection

The historical recorder records data gathered by the PAS in historical data sets at regular intervals.

The historical recorder checks each data set’s status (open, active, quiesced, unquiesced, error, closed) and recording eligibility (yes or no) to determine which data set to write to.

When the historical recorder must search for a data set to write to, the search starts with the next data set in numerical order—in the case of the previous example, the data set named on the HISTDS04 DD statement. If the data set named on HISTDS04 is not empty, the historical recorder keeps searching for an empty data set by using data sets defined on the HISTDSnn DD statements.
If an allocated, empty data set does not exist on any of the HISTDSnn DD statements, the recorder selects the data set containing the oldest data and overwrites it.

**Forced file selection**

Although the historical recorder selects the *first* historical data set to write to, you have control over the subsequent data sets that are selected for data recording by using line commands from the DSLIST view.

You can use these line commands to perform all historical database maintenance (with the exception of archival and retrieval of historical data sets).

See “Managing the historical database” on page 215 for more information.

**How many historical data sets should you allocate?**

You can allocate as many or as few data sets as you need, up to the limit of 100.

Default customization provides for 3 data sets. When a data set is full, the historical recorder uses the next data set until that one is also full, and so on. When all the data sets are full, recording begins again with the first data set and the data is overwritten.

To determine the appropriate number of historical data sets for your site, collect sample data for one hour and then allocate the data sets according to how many hours worth of data you want to collect.

*Note*

You can use the DSLIST view to add historical data sets dynamically, as described in “Adding a historical data set dynamically” on page 225.

**Allocating historical data sets to the historical database**

A historical data set is added to DSLIST by being allocated to the PAS as part of the historical database.

After you have created and formatted the VSAM files that comprise the historical database as described in the *MainView Customization Reference*, you are ready to allocate the data sets that you want the historical recorder to write to.
MainView provides two methods for allocating historical data sets to a PAS, as shown in Table 19 on page 214.

**Table 19: Historical database allocation methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic allocation through the DSLIST view</td>
<td>“Adding a historical data set dynamically” on page 225</td>
</tr>
<tr>
<td><strong>Note:</strong> The PAS does <em>not</em> need to be recycled if you add historical data sets by using this method.</td>
<td></td>
</tr>
<tr>
<td>Allocation at PAS initialization by including DD statements in the PAS started task procedure for each data set</td>
<td><em>MainView Customization Reference</em></td>
</tr>
<tr>
<td><strong>Note:</strong> The PAS <em>must</em> be recycled if you add historical data sets by using this method.</td>
<td></td>
</tr>
</tbody>
</table>

If you prefer, you can use a combination of both methods (hardcoding DD statements in the PAS procedure and interactive allocation with the DSLIST view) to allocate as many historical data sets as you need during your user session.

### Defining historical data support

This procedure describes how to define historical data support for your product.

**To define historical data support for your product**

1. Allocate and format a historical database of VSAM data sets for the PAS supporting your products (see “Understanding the historical database” on page 209).

2. Define the data sets in the historical database to the PAS started task procedure (see your product’s customization guide) or use the DSLIST view (see “Allocating historical data sets to the historical database” on page 213).

**Note**

BMC recommends that you define data sets to the historical data set by using the DSLIST view because this method provides more flexibility with initial recording options and deallocation.

After the historical database is created and at least one file is allocated to hold historical data, you manage the historical database and control access to historical data by using the DSLIST view. DSLIST allows you to perform tasks such as:

- determining which historical data sets are available for recording data
■ changing the status of historical data sets

■ controlling read and write access to the historical data sets

You also might want to archive and restore historical data sets with the standard utility your site uses.

**Managing the historical database**

This topic covers the maintenance tasks you perform to control historical data and manage the historical database that is used by a PAS.

To control the historical database for the current PAS, use the tasks in Table 20 on page 215.

**Table 20: Historical database management tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>View or Command</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understand how the historical database is structured and how data is recorded to the historical database</td>
<td>Not applicable</td>
<td>“Understanding the historical database” on page 209</td>
</tr>
<tr>
<td>View the status of all historical data sets</td>
<td>DSLIST view</td>
<td>“Viewing the status of historical data sets (DSLIST view)” on page 216</td>
</tr>
<tr>
<td>Understand what the status of a historical data set means for recording or displaying data</td>
<td>Online Help for the Status field on the DSLIST view</td>
<td>“Understanding the status of a historical data set” on page 217</td>
</tr>
<tr>
<td>Change a data set’s recording eligibility</td>
<td>O line command</td>
<td>“Changing recording eligibility” on page 218</td>
</tr>
<tr>
<td>Select the next data set for recording</td>
<td>F line command</td>
<td>“Selecting the next data set for recording” on page 218</td>
</tr>
<tr>
<td>Restrict read and write access to a data set</td>
<td>Q line command</td>
<td>“Restricting read and write access to the active data set” on page 220</td>
</tr>
<tr>
<td>Reinststate read and write access to a quiesced data set</td>
<td>U line command</td>
<td>“Reinstating read and write access to a quiesced data set” on page 222</td>
</tr>
<tr>
<td>Close a data set</td>
<td>F or Q line command</td>
<td>“Closing a data set” on page 223</td>
</tr>
<tr>
<td>Empty a data set</td>
<td>R line command</td>
<td>“Emptying a historical data set” on page 224</td>
</tr>
</tbody>
</table>
### Viewing the status of historical data sets (DSLIST view)

Use the DLIST view to manage the historical database.

DSLIST shows the names of the data sets that comprise the historical database and allows you to issue commands against these data sets. Using DLIST, you can allocate, deallocate, empty, or close a data set and control the read and write access to a data set.

**To access the DLIST view**

1. Access a MainView product that uses historical data and ensure that the product is running in windows mode.

   **Note**
   
Plex Manager does not contain a DLIST view because it does not use historical data.

2. On the **COMMAND** line, type **DSLIST** and press **Enter**.

   **Note**
   
   For CMF MONITOR version 5.8.00 or later and MainView for z/OS, display the DLISTZ view first. Then hyperlink on the type of historical data set (regular, long-term, or short-term) for which you want information.

The DLIST view is displayed, as shown in **Figure 55 on page 216**.

**Figure 55: DLIST view**

ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) -----------
COMMAND ===> SCROLL ===> PAGE
CURR WIN ===> 1 , ALT WIN ===> >W1 =DSLIST============SYSB=====*=======dd/mm/yyyy=hh:mm:ss====MVMVS==D=====3

### Task View or Command See

<table>
<thead>
<tr>
<th>Task</th>
<th>View or Command</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocate a historical data set to the PAS</td>
<td>ADD primary command or PAS started task procedure</td>
<td>“Adding a historical data set dynamically” on page 225 or “Allocating historical data sets to the historical database” on page 213</td>
</tr>
<tr>
<td>Permanently save changes made to historical data sets</td>
<td>SAVE primary command</td>
<td>“Saving dynamic changes to the historical database” on page 227</td>
</tr>
<tr>
<td>Deallocate a historical data set</td>
<td>D line command</td>
<td>“Deallocating a historical data set” on page 227</td>
</tr>
<tr>
<td>Move data to and from the archive</td>
<td>REPRO or EXPORT function in IDCAMS</td>
<td>“Archiving and retrieving historical data” on page 228</td>
</tr>
</tbody>
</table>
The DSLIST view displays information about historical data sets that are currently allocated to the PAS. For information about any field in the DSLIST view, place the cursor on the field and press the Help key.

### Understanding the status of a historical data set

The **Status** field indicates the current state of a historical data set.

**Table 21 on page 217** shows the valid **Status** field values that can appear for a data set on the DSLIST view and explains what each value means in terms of reading or writing availability.

#### Table 21: Understanding the status of a historical data set

<table>
<thead>
<tr>
<th>Status</th>
<th>Available for writing</th>
<th>Available for reading</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Active</strong></td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Only one data set at a time per PAS can have an Active status.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Closed</strong></td>
<td>Yes, if the <strong>Rec</strong> field indicates Yes no, if the <strong>Rec</strong> field indicates No</td>
<td>Yes, regardless of the <strong>Rec</strong> field value no, if the <strong>Pending</strong> field value is <strong>SEL PEND</strong></td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>There is an error condition with the data set.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Open</strong></td>
<td>No</td>
<td>YES</td>
</tr>
<tr>
<td>This data set was the most recent Active data set before the historical recorder switched to a different data set, but this data set has not been closed yet.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The Open status rarely occurs; it could indicate an impending error condition for the data set.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Qscd</strong></td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Changing recording eligibility

To change a historical data set's recording eligibility, use the O line command.

This command can be used on data sets with a Closed status only; it has no effect on data sets with another status.

The recording eligibility, originally set on the Add Historical Data Set dialog, appears in the Rec field and indicates whether the historical recorder can write data to a data set.

The O line command controls write access to a data set and toggles recording eligibility from Yes to No and back again each time it is issued against a data set.

Figure 56 on page 218 shows how to issue the O line command against a data set to control write access by the historical recorder.

Figure 56: Using the O line command to change recording eligibility

| ddmmmyyyy hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) ------- |
| COMMAND ===>                           SCROLL ===> PAGE |
| CURR WIN ===> 1        ALT WIN ===> |
| >W1=DSLIST===========SYSB=====*=======dd/mm/yyyy=hh:mm:ss====MVMVS=D======3 |
| C DDNAME   From Date  Time  To Date    Time  Rec Status Pending Data set nam |
| - ------   ---------- ----- ---------  ----- --- ------ ------- ------------ |
| HISTDS03 dd/mm/yyyy hh:mm dd/mm/yyyy hh:mm Yes Active ***** PROD1.IMAGSY |
| o HISTDS01 dd/mm/yyyy hh:mm dd/mm/yyyy hh:mm No Closed ***** PROD1.IMAGSY |
| HISTDS02 dd/mm/yyyy hh:mm dd/mm/yyyy hh:mm Yes Closed ***** PROD1.IMAGSY |

When you press Enter, the value in the Rec field immediately changes to Yes or No.

Selecting the next data set for recording

To force the historical recorder to switch recording to a different data set at the end of the current interval, issue the F line command.

You can issue the F line command against only one data set per interval. After the command is issued, you must wait until the end of the current interval for command processing to be completed before you can issue this command against a different data set.

The F line command can be used only against a historical data set that has a Status field value of Closed and a Rec field value of Yes; it has no effect on data sets with any other status.

- If the historical data set that you want to select has a Rec field value of No, issue the O line command to change its eligibility to Yes; see “Changing recording eligibility” on page 218.
If the historical data set that you want to select has a Status field value other than Closed, see “Closing a data set” on page 223.

When the F line command is issued, the Pending field value changes to SEL PEND for the selected data set and to QUIESCED for the currently active data set.

When the current interval ends, results are as follows:

- The historical recorder switches recording to the selected data set and writes the record for the interval that just ended.

- The Status field changes to Active and the Pending field value changes back to asterisks for the selected data set.

- The Status field changes to Qscd and the Pending field value retains a QUIESCED status for previously active data set.

**Note**

The previously active data set retains a QUIESCED pending state throughout the duration of the new interval. Data cannot be written to or read from this data set from the time the F line command is issued until the end of the new interval.

At the end of the new interval, the QUIESCED pending state ends automatically and the Pending field value changes to a Qscd status. A quiesced data set must be manually unquiesced to become eligible for recording or reading again. Issue the U line command to unquiesce a quiesced data set; see “Reinstating read and write access to a quiesced data set” on page 222.

Figure 57 on page 219 shows how to issue the F line command against a data set to force the historical recorder to switch recording to it at the end of the current interval.

**Figure 57: Using the F line command to switch data set recording**

<table>
<thead>
<tr>
<th>ddmmyyyy</th>
<th>hh:mm:ss</th>
<th>---</th>
<th>MainView WINDOW INTERFACE (Vvm.mm)</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>==&gt;</td>
<td>SCROLL</td>
<td>==&gt;</td>
<td>PAGE</td>
</tr>
<tr>
<td>CURR WIN</td>
<td>==&gt; 1</td>
<td>ALT WIN</td>
<td>==&gt;</td>
<td>&gt;W1 =DSLIST=SYSB==*==dd/mm/yyyy=hh:mm:ss==MVMVS=D==3</td>
</tr>
<tr>
<td>DDNAME</td>
<td>From Date</td>
<td>To Date</td>
<td>Time</td>
<td>Rec</td>
</tr>
<tr>
<td>HISTDS03</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
<tr>
<td>HISTDS01</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
<tr>
<td>HISTDS02</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
</tbody>
</table>

Figure 58 on page 219 shows how the Pending field changes when you issue the F line command.

**Figure 58: Pending field changes for the forced and active data sets**

<table>
<thead>
<tr>
<th>ddmmyyyy</th>
<th>hh:mm:ss</th>
<th>---</th>
<th>MainView WINDOW INTERFACE (Vvm.mm)</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>==&gt;</td>
<td>SCROLL</td>
<td>==&gt;</td>
<td>PAGE</td>
</tr>
<tr>
<td>CURR WIN</td>
<td>==&gt; 1</td>
<td>ALT WIN</td>
<td>==&gt;</td>
<td>&gt;W1 =DSLIST=SYSB==*==dd/mm/yyyy=hh:mm:ss==MVMVS=D==3</td>
</tr>
<tr>
<td>DDNAME</td>
<td>From Date</td>
<td>To Date</td>
<td>Time</td>
<td>Rec</td>
</tr>
<tr>
<td>HISTDS03</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
<tr>
<td>HISTDS01</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
<tr>
<td>HISTDS02</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
</tr>
</tbody>
</table>
Figure 59 on page 220 shows how the Status field changes at the end of the current interval. Also note that the data sets are reordered because they are sorted on the From Date and Time fields.

**Figure 59: End-of-interval status field changes**

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>From Date</th>
<th>Time</th>
<th>To Date</th>
<th>Time</th>
<th>Rec</th>
<th>Status</th>
<th>Data set</th>
</tr>
</thead>
<tbody>
<tr>
<td>HISTDS01</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>Yes</td>
<td>Active</td>
<td>PROD1.IMAG</td>
</tr>
<tr>
<td>HISTDS03</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>Yes</td>
<td>QUIESCED</td>
<td>PROD1.IMAG</td>
</tr>
<tr>
<td>HISTDS02</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>No</td>
<td>Closed</td>
<td>PROD1.IMAG</td>
</tr>
</tbody>
</table>

Figure 60 on page 220 shows how the Pending field changes at the end of the new interval.

**Figure 60: Pending field change**

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>From Date</th>
<th>Time</th>
<th>To Date</th>
<th>Time</th>
<th>Rec</th>
<th>Pending</th>
<th>Data set</th>
</tr>
</thead>
<tbody>
<tr>
<td>HISTDS01</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>Yes</td>
<td>Active</td>
<td>PROD1.IMAG</td>
</tr>
<tr>
<td>HISTDS03</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>Yes</td>
<td>QUIESCED</td>
<td>PROD1.IMAG</td>
</tr>
<tr>
<td>HISTDS02</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>dd/mm/yyyy</td>
<td>hh:mm</td>
<td>No</td>
<td>Closed</td>
<td>PROD1.IMAG</td>
</tr>
</tbody>
</table>

**Restricting read and write access to the active data set**

To restrict read and write access to the active historical data set at the end of the interval, issue the Q line command.

After the Q line command is issued, you can neither write to nor read from the active data set and you must wait until the end of the current interval for command processing to be completed.

The Q line command can be used against a historical data set that has a Status field value of Active only; it has no effect on data sets with another status.

When the Q line command is issued against the active data set, the Pending field value changes to QUIESCED. Recording to or reading from the active quiesced data set is prohibited immediately.

When the current interval ends, the results are as follows:

- The historical recorder switches recording to the next available data set and writes the record for the interval that just ended.
The **Status** field changes to Active for the new data set.

The **Status** field changes to Qscd for the previously active data set and the **Pending** field retains a QUIESCED value.

---

**Note**

The previously active data set retains a QUIESCED pending state throughout the duration of the new interval. Data cannot be written to or read from this data set from the time the Q line command is issued until the end of the new interval.

At the end of the new interval, the QUIESCED pending state automatically ends and the **Pending** field value changes to a Qscd status. A quiesced data set must be manually unquiesced to become eligible for recording or reading again. Issue the U line command to unquiesce a quiesced data set; see “Reinstating read and write access to a quiesced data set” on page 222.

---

**Figure 61 on page 221** shows how to issue the Q line command against the active data set to force the historical recorder to switch recording to the next available data set at the end of the current interval.

**Figure 62 on page 221** shows how the **Pending** field changes when you issue the Q line command.

**Figure 63 on page 221** shows how the **Status** field changes at the end of the current interval.
Even though the HISTDS02 data set has older data than the HISTDS01 data set, the HISTDS01 data set becomes the Active data set because it has a Rec field value of Yes. The HISTDS02 has a Rec field value of No, so it is not eligible to have data written to it; see “Changing recording eligibility” on page 218 for more information.

Figure 64 on page 222 shows how the Pending field changes at the end of the new interval.

Figure 64: Pending field change at the end of the new interval

Reinstating read and write access to a quiesced data set

To reinstate read and write access to a quiesced historical data set, issue the U line command.

The U line command can be used against a historical data set that has a Status field value of Qscd and a Pending field value of asterisks. It has no effect on data sets with another status.

Note

A QUIESCED pending state is retained on a quiesced data set for the period of one interval. When the interval ends, the pending state automatically changes to time.

A data set is placed in a Qscd status when the F line command or Q line command is issued against it. See “Selecting the next data set for recording” on page 218 and “Restricting read and write access to the active data set” on page 220 for more information about these commands.

When the U line command is issued against a data set with a Qscd status and a Pending field value of asterisks, the data set is closed immediately and the Status field value changes to Closed.
Figure 65 on page 223 shows how to issue the U line command against a quiesced data set to reinstate read and write access.

**Figure 65: Using the U line command to reinstate read/write access**

<table>
<thead>
<tr>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>W1 DSLIST</td>
<td>“Selecting the next data set for recording” on page 218 or “Restricting read and write access to the active data set” on page 220 and “Reinstating read and write access to a quiesced data set” on page 222</td>
</tr>
<tr>
<td>U</td>
<td>“Changing recording eligibility” on page 218</td>
</tr>
</tbody>
</table>

### Closing a data set

There are a number of ways to close a data set, depending on the data set's current status.

A closed status for a data set is important because it enables a data set for reading. The ability to write to a closed data set is controlled by the O line command; see “Changing recording eligibility” on page 218.

Also, a data set must be closed before the following historical management tasks can be performed:

- “Changing recording eligibility” on page 218
- “Selecting the next data set for recording” on page 218
- “Emptying a historical data set” on page 224
- “Deallocating a historical data set” on page 227

Table 22 on page 223 describes how to close a data set for each Status field value.

**Table 22: Closing a historical data set**

<table>
<thead>
<tr>
<th>Status</th>
<th>Command</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>F or Q line command to change the Active status to Qscd</td>
<td>“Selecting the next data set for recording” on page 218 or “Restricting read and write access to the active data set” on page 220 and “Reinstating read and write access to a quiesced data set” on page 222</td>
</tr>
<tr>
<td></td>
<td>U line command to change the status from Qscd to Closed</td>
<td></td>
</tr>
<tr>
<td>Closed</td>
<td>O line command to set recording eligibility</td>
<td>“Changing recording eligibility” on page 218</td>
</tr>
<tr>
<td>Status</td>
<td>Command</td>
<td>Reference</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>Error</td>
<td>R line command to make the data set eligible for recording again</td>
<td>“Emptying a historical data set” on page 224</td>
</tr>
<tr>
<td></td>
<td>The R line command empties the data in the data set. If you can determine a reason for the error condition based on diagnostic messages, you may be able to archive the data set before resetting it.</td>
<td></td>
</tr>
<tr>
<td>Open</td>
<td>None</td>
<td>Operator's console or job log for messages that might indicate a reason why the data set cannot close</td>
</tr>
<tr>
<td></td>
<td>The data set is pending a closed condition and should close automatically unless an error condition is present. An Open status rarely occurs. If this status is retained for more than one interval, an I/O error could be keeping the data set from closing.</td>
<td></td>
</tr>
<tr>
<td>Qscd</td>
<td>U line command to change the Qscd status to Closed</td>
<td>“Reinstating read and write access to a quiesced data set” on page 222</td>
</tr>
</tbody>
</table>

### Emptying a historical data set

To reset, or empty, a historical data set and make it available for reuse, issue the R line command.

**Note**

Offload the data into an inactive historical data set or to tape *first* if you want to save it for later use. See “Archiving and retrieving historical data” on page 228 for more information.

The R line command can be used against a historical data set that has a **Status** field value of Closed or Error; it has no effect on data sets with another status.

**Figure 66 on page 224** shows how to issue the R line command against a closed data set to empty a historical data set.

**Figure 66: Using the R line command to empty a historical data set**
When the R line command is issued, the Reset Confirm dialog, shown in Figure 67 on page 225, is displayed for confirmation that you want to discard the data.

**Figure 67: Confirming that a historical data set should be emptied**

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>SCROLL ===&gt; PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reset Confirm</td>
<td></td>
</tr>
<tr>
<td>Confirm the resetting of the Interval Recorder dataset</td>
<td></td>
</tr>
<tr>
<td>Dataset name: PROD1.IMAGSYSB.HISTDS02</td>
<td></td>
</tr>
<tr>
<td>Confirm reset of dataset: (Yes/No)</td>
<td></td>
</tr>
<tr>
<td>END to exit resetting historical dataset</td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit without resetting historical dataset</td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
</tr>
</tbody>
</table>

To empty the data set, type **Yes** and press END. To cancel the reset command, type **No** or leave the field blank.

When END is entered, the DSLIST view is redisplayed. Figure 68 on page 225 shows how the empty data set is displayed on the DSLIST view.

**Figure 68: How an empty historical data set is displayed**

| ddmmmyyyy | hh:mm:ss | -------- | MainView | WINDOW | INTERFACE | (Vv.r.mm) | -------- |
|------------|----------|----------|----------|--------|-----------|-----------|
| COMMAND ===> | SCROLL ===> PAGE |
| C C C C W1 =DSLIST==ateau**SYSB====*====**dd/mm/yyy=hh:mm:ss==MVMVS==D====3 | |
| DDNAME | From Date | Time | To Date | Time | Rec Status | Pending | Data set nam |
| HISTDS01 | dd/mm/yyyy | hh:mm | dd/mm/yyyy | hh:mm | Yes | Active | ****** | PROD1.IMAGSYSY |
| HISTDS02 | dd/mm/yyyy | hh:mm | dd/mm/yyyy | hh:mm | Yes | Qscd | ****** | PROD1.IMAGSYSY |
| HISTDS03 | 00:00 | 00:00 | No | Closed | ****** | PROD1.IMAGSYSY |

**Note**

If the **Rec** field indicates No for an empty data set, use the O line command to enable this data set for recording; see “Changing recording eligibility” on page 218.

---

**Adding a historical data set dynamically**

To allocate a historical data set dynamically so that it is added to the historical database and appears in DSLIST, enter the ADD command.

The ADD command can be entered as a primary command or as a line command.

When you use the ADD command, you have a choice of permanently or temporarily allocating the historical data set. To permanently save a dynamically allocated historical data set so that it is retained when the PAS is recycled, issue the SAVE command while adding a historical data set; otherwise, the data set is allocated temporarily.
To add a historical data set dynamically

1. Create the data set by following the instructions for your type of PAS (z/OS or BBI-SS) in the MainView Customization Reference.

2. Display the DSLIST view for the product for which you want to add a historical data set.

3. Enter the ADD command in one of the following ways:
   - On the COMMAND line, type `ADD` and press Enter.
   - In the line command column, type `A` and press Enter.

   The Add Historical Dataset dialog is displayed, as shown in Figure 69 on page 226.

   **Figure 69: Add Historical Dataset dialog**

<table>
<thead>
<tr>
<th>COMMAND ===&gt;</th>
<th>Add Historical Dataset</th>
<th>SCROLL ===&gt;</th>
<th>CSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter dataset name and recording eligibility</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Enter a fully qualified dataset name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Do not enclose dataset name in quotes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Dataset must have already been created</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dataset name:</td>
<td>May be used for recording? Yes/No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END to exit adding historical dataset</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CANCEL to exit the add dialog</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELP to view related help</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET to reset to initial data</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. Enter the name of the historical data set that you want to allocate in the Dataset name field.

5. Specify whether you want this data set to be eligible for recording by typing Yes or No in the Record? field and then pressing Enter.

   The message DATA SET STARTED is displayed.

   **Note**

   Use this field to set recording eligibility of a data set initially; use the O line command to control recording eligibility after the data set is allocated (see “Changing recording eligibility” on page 218 for more information).

6. *(optional)* Add information for additional data sets, if necessary, and press Enter for each data set to add it dynamically.

7. Press END to return to the DSLIST view.
The message **DATA SET ALLOCATED** is displayed for each newly allocated data set and the data set name is added to the DLIST view. Dynamic allocation is complete but does not make the data set a permanent member of the historical database. When the PAS is recycled, the data set name no longer appears on DLIST and must be added again.

To permanently add a dynamically allocated data set to the historical database, issue the **SAVE** command.

---

**Saving dynamic changes to the historical database**

To permanently save a dynamically added historical data set to the historical database, or to permanently remove a deallocated data set (that was originally added dynamically) from the historical database, issue the **SAVE** command after allocating or deallocating the data set.

When you use the **SAVE** command, you affect the contents of the PARMFILE DD data set. The data set names that are currently displayed on the DLIST view are saved to the PARMFILE data set and become part of the historical database.

Any historical data set names in PARMFILE are permanently part of the historical database because when the PAS is initialized, both the HISTDSnn DD and the PARMFILE DD statements are read—the data sets defined to these statements make up the historical database at PAS initialization.

Therefore, if you add a historical data set and issue **SAVE**, it becomes part of the historical database, and if you deallocate a historical data set that was dynamically added and issue **SAVE**, the data set is removed from the historical database.

The **SAVE** command is issued on the **COMMAND** line.

If you do not issue the **SAVE** command, historical data sets that are dynamically added are lost from, and those that are deallocated are added back to, the historical database at the next PAS initialization.

---

**Deallocating a historical data set**

To remove a historical data set that was dynamically allocated from the historical database and remove its name from the DLIST view, issue the **D** line command.

The **D** line command can be used against a historical data set that has a **Status** field value of **Closed**; it has no effect on data sets with another status or on data sets that are allocated to the historical database through a HISTDSnn DD DD statement.
Permanent deallocation occurs when you deallocate a data set that was dynamically added and then issue the SAVE command.

Temporary deallocation occurs if you issue only the D line command against a data set and not the SAVE command. The exception to this case is when a data set was added and saved during a previous session. The data set name appears in PARMFILE and is read and allocated as part of the historical database at the next PAS initialization.

Issue the SAVE command to permanently remove a data set name from PARMFILE and permanently deallocate the data set from the historical database.

Figure 70 on page 228 shows how to issue the D line command against a closed data set to remove the historical data set from the DSLIST view.

After you press Enter, the data set is deallocated and removed from the DSLIST view. You can reallocate a historical data set again by using the ADD command.

Archiving and retrieving historical data

You can use the REPRO or EXPORT function of the IBM IDCAMS utility or the standard utility that your site uses to archive historical data to tape or to offline data sets.

At a later time, you can use the same utility to retrieve the data from the archive into a historical data set for viewing.

Archiving data

As soon as a historical data set fills up, MainView sends a status message to the operator console. BMC recommends setting up an automation product that uses this operator console message to trigger the IDCAMS REPRO or EXPORT function to offload historical data automatically to the archive each time a data set reaches capacity. For example, you can use one of these automation products:

- MainView AutoOPERATOR
- Standard utility that your site uses
Retrieving data

When you restore archived historical data, the data set that you move the data to must be currently allocated and must not be eligible for recording.

When you complete the restoration of a historical data set from tape, either no time and date stamps are displayed or invalid date and time stamps are displayed in the From Date, To Date, and Time fields for the data set.

The date and time data is not correct because the PAS reads the date and time of historical data sets only at data set initialization time or when data sets are added dynamically.

To restore archived historical data and display the correct date and time of a restored data set

1. Use the D line command to deallocate the historical data set.
2. Restore data from the archive by using an IDCAMS or other standard utility.
3. Use the ADD command to add the historical data set to the historical database.
4. (optional) Use the SAVE command if you want to permanently save the data set allocations so that they are not removed at the next PAS initialization.
Managing the MainView Infrastructure common registry

This chapter explains how to mount, identify, activate, and manage a MainView Infrastructure common registry file system.

Note
In previous releases, the file system for the MainView Infrastructure common registry was known as the Alarm Management file system.

The MainView Infrastructure common registry is used to store information for such features as Alarm Management, deployment, and user preferences.

The common registry requires a UNIX System Services (USS) HFS or zFS file system. The file system is allocated during customization by using the Installation System or manually following the procedure in the MainView Customization Reference.

The common registry must also be mounted to the operating systems and identified to the CASs.

You can have one or more common registries: one per CAS or one that is shared among the CASs in a sysplex.

Mounting a file system

To mount a common registry file system on an operating system, you must add a MOUNT statement to the BPXPRMxx member in SYS1.PARMLIB.

The MOUNT statement will mount the file system on subsequent IPLs.

BMC recommends that the file system be mounted off a user mount point or the IBM /var mount point. You must create a directory to use as a mount point for the common registry file system. Do not choose an existing directory that is used for some other purpose.
WARNING

- Mounting a file system on an existing directory will make all the existing data inaccessible.
- If the mount point is the system root, the common registry will not initialize.

BMC has no dependency on the location of this directory. The best location will depend on your site’s configuration and use of the HFS or zFS file system.

**Identifying a file system to a CAS**

To identify a common registry file system to a CAS you must tell the CAS where the mount point is by using the HFSPATH view in Plex Manager.

*Note*

In previous releases, the file system for the MainView Infrastructure common registry was known as the Alarm Management file system. The file system was identified to a CAS by using the BBMHFS00 statement in a CAS started task procedure.

If you had a file system identified to a CAS by using the BBMHFS00 statement, you must perform the task in this topic for the CAS to know about the file system. The BBMHFS00 statement is no longer supported by MainView Infrastructure.

The HFSPATH view displays entries in the Runtime Component System(RTCS) system registry that pertain to the common registry file system. An entry contains the path to the file system and specifies which CAS should use the entry.

You can identify a unique file system for a CAS to use, or you can define a single shared file system that is used by multiple CASs. A unique entry is given the same name as the related CAS. The shared entry is named $SHARED$.

When a CAS starts, it looks for a unique entry in the RTCS system registry to use. If there is no unique entry for the CAS, the CAS uses the shared entry.

**To identify a file system to a CAS**

1. Access the HFSPATH view as described in “Accessing the HFSPATH view” on page 234.
2. Place the cursor next to an entry you want to use as a model in the CMD column, type ADD, and press Enter.

The Define or Add a New HFSPATH dialog is displayed.
3 Enter the name of the CAS that is going to use the path data specified in this entry, or leave the field blank if the entry is going to be shared by the CASs in the SSI.

4 Change the HFS path as needed. The first 60 characters of the path are entered in the HFSPATH field. The remaining 68 characters are entered in the path name continued field (...........).

5 Enter the END command to save the new entry.

Activating a file system

A file system must be active to be used by a CAS.

To activate a file system

1 Access the HFSPATH view as described in “Accessing the HFSPATH view” on page 234.

2 Place the cursor in the CMD column next to the entry you want to activate, type ACT, and press Enter.

   The Activate HFSPATH for a CAS dialog is displayed.

3 Enter the END command to activate the entry.

Other file system tasks

In addition to identifying a file system to a CAS and activating a file system, you can perform the following actions:

- Change the file system path by using the CHAnge line command.
- Delete a file system path definition by using the DEL line command.
- Inactivate a file system path by using the INActivate line command.

See the online Help for additional information.
To access the HFSPATH view, perform one of the following actions:

- On the EZPLEX menu under **Administration**, position the cursor on **HFSPATH entries** and press **Enter**.

- In Plex Manager, type **HFSPATH** on the **COMMAND** line and press **Enter**.

- From any MainView product that is running in windows mode, use the following **CONtext** command to access Plex Manager and display the HFSPATH view:

  CON * PLEXMGR;HFSPATH

The HFSPATH view is displayed, as shown in Figure 71 on page 234.

The entry that is in use by the local CAS is the one with a status of Active.
Managing the User Interface Middleware (UIM) server

The User Interface Middleware (UIM) server, which resides on the mainframe, provides web services support for BMC products and handles communication for console-enabled products.

The Installation System installs the UIM server libraries if you select:

- Products that require web services support
- Console-enabled products, or products with a console-enabled feature
- MainView Infrastructure environment

This chapter explains how to perform the following tasks:

- “Creating a UIM server” on page 235
- “Configuring the UIM server” on page 243
- “Operating the UIM server” on page 252
- “Troubleshooting UIM server problems” on page 258

Creating a UIM server

When you install and customize the UIM server, the Installation System creates the following sample members:

<table>
<thead>
<tr>
<th>Member</th>
<th>Saved in</th>
</tr>
</thead>
<tbody>
<tr>
<td>Startup configuration member</td>
<td>UIM Customized Config Library</td>
</tr>
<tr>
<td>Startup procedure</td>
<td>UIM Customized Parm Library</td>
</tr>
</tbody>
</table>
If you want to create additional UIM servers, perform the following tasks:

1. “Creating a startup configuration member” on page 236
2. “Creating a started task procedure” on page 237
3. “Defining the MainView Infrastructure CAS connection (optional)” on page 239
4. “Setting up the HFS data set” on page 240

Creating a startup configuration member

Use the following procedure to create a startup configuration member for the new UIM server.

To create a startup configuration member

1. Locate one of the following members:
   - #NORMAL in the UIM sample library
   - Customized startup configuration member in the UIM Customized Config Library

2. Copy the member to your configuration file and give it a new name.

   Figure 72 on page 236 shows the sample startup configuration member. The HLQ value is the high-level qualifier that you specified during installation.

   Figure 72: Startup configuration member

```
<BMCHTTP>
   <BMC_PARM ID="PORT" VALUE="3683" />
   <BMC_PARM ID="AUTH_TIMEOUT_SECS" VALUE="1800" />
   <BMC_PARM ID="AFF_TIMEOUT_SECS" VALUE="1800" />
   <BMC_PARM ID="HFS_DATASET" VALUE="HLQ.HFS" />
   <BMC_PARM ID="ALLOW_NETCMD" VALUE="YES" />
   <BMC_PARM ID="ENCRYPTION_LEVEL" VALUE="CREDENTIALS-IF" />
   <BMC_PARM ID="USS_HFS_ROOT" VALUE="/tmp/bmcuim" />
</BMCHTTP>
```

3. Edit the new startup configuration member by changing the parameters listed in Table 23 on page 237.
Table 23: Startup configuration member parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
<th>Accepted values</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;BMC_PARM ID=&quot; PORT&quot; VALUE=&quot;3683&quot;/&gt;</td>
<td>Port number for the UIM server</td>
<td>Unique numeric value between 1 and 65535 The default is 3683.</td>
<td>“Changing the port number” on page 245</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; AUTH_TIMEOUT_SECS&quot; VALUE=&quot;1800&quot;/&gt;</td>
<td>Security authorization timeout for a console</td>
<td>Numeric value in seconds The default is 1800.</td>
<td>“Changing the security authorization timeout feature permanently” on page 245</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; AFF_TIMEOUT_SECS&quot; VALUE=&quot;1800&quot; /&gt;</td>
<td>Idle timeout period for affinity tasks</td>
<td>Numeric value in seconds The default is 1800.</td>
<td>“Changing the idle timeout for affinity tasks” on page 246</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; HFS_DATASET&quot; VALUE=&quot; HLQ.HFS &quot; /&gt;</td>
<td>HFS data set name</td>
<td>PDSE data set name for storing HFS data</td>
<td>“Changing the HFS server-side storage data set name” on page 246</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; ALLOW_NETCMD&quot; VALUE=&quot;YES&quot; &quot;&gt;</td>
<td>Whether to enable the browser command interface</td>
<td>YES (default), NO, or AUTHORIZE</td>
<td>“Enabling or disabling the network browser command interface” on page 247</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; ENCRYPTION_LEVEL&quot; VALUE=&quot;CREDENTIALS-IF&quot; &quot;&gt;</td>
<td>Whether to enable encryption of user credentials</td>
<td>CREDENTIALS-IF (default) or NO</td>
<td>“Enabling or disabling encryption of user credentials” on page 247</td>
</tr>
<tr>
<td>&lt;BMC_PARM ID=&quot; USS_HFS_ROOT&quot; VALUE=&quot;/tmp/bmcuim&quot; /&gt;</td>
<td>Root directory in UNIX System Services (USS) HFS for web service configuration members</td>
<td>/tmp/bmcuim (default)</td>
<td>“Setting the USS HFS root directory” on page 248</td>
</tr>
</tbody>
</table>

4 Save the startup configuration member.

Creating a started task procedure

After you create a startup configuration member for the UIM server, use the following procedure to create a started task procedure for the server.
To create a started task procedure

1. Locate one of the following members:
   - **#UIMX** in the UIM sample library
   - Customized started task procedure in the UIM Customized Parm Library

2. Copy the member to your system procedure library and give it a new name.

   Figure 73 on page 238 shows the sample #UIMX member.

**Figure 73: #UIMX member**

```plaintext
//uimx     PROC M=uimx,                <----- NAME OF CONFIGURATION MEMBER
//             ENV=
//     *------------------------------------------------------------------
//uimx     EXEC PGM=UIMMAIN,               +
//             ACCT=(acct),        <--- SPECIFY ACCOUNTING INFO         +
//             REGION=OK,         <--- SPECIFY REGION SIZE              +
//             TIME=1440,                                              +
//             PARM=('-C &M &ENV -L =B =CNFTRACE =VERSION')
//     *------------------------------------------------------------------
//             -C MMMMMM CONFIGURATION FILE MEMBERNAME
//             -P 3683 TCP LISTENER PORT NUMBER
//             -L LOG MESSAGES AND TRACE VIA SUBTASK
//     *------------------------------------------------------------------
//             =SOUT= SPECIFY THE SYSOUT CLASS FOR DYNAMICALLY ALLOCATED
//             LOG FILES( IE. =SOUT= X )
//     *------------------------------------------------------------------
//             =CNFTRACE PRINT DIAGNOSTICS DUE TO TCP/IP CONFIGURATION FAILURES
//             =VERSION PRINT RUNTIME LIBRARY RELEASE INFORMATION TO SYSTERM
//     *------------------------------------------------------------------
//STEPLIB  DD  DISP=SHR,DSN= bmc.uim.load
//         DD  DISP=SHR,DSN= bmc.pch.load
//SYSTERM  DD  SYSOUT=*                
//SYSPRINT DD  SYSOUT=*                
//STGRPT   DD  SYSOUT=*                
//MSGLOG   DD  SYSOUT=*                
//TRCLOG   DD  SYSOUT=*                
//HTPCONT  DD  DISP=SHR,DSN= bmc.uim.content
//HTPPARM  DD  DISP=SHR,DSN=HLQ,customized.config
//         DD  DISP=SHR,DSN= bmc.uim.config
```

Creating a UIM server
3 Edit the new #UIMX member by changing the variables listed in Table 24 on page 239.

Table 24: #UIMX member variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
<th>Accepted value</th>
</tr>
</thead>
<tbody>
<tr>
<td>uimx</td>
<td>Name of the started task procedure and the startup configuration member</td>
<td>#UIMX member name that you specified when copying the member to your system procedure library (Step 2 on page 238)</td>
</tr>
<tr>
<td>bmc.uim.load</td>
<td>Library that contains the UIM server and product execution code</td>
<td>Valid data set name</td>
</tr>
<tr>
<td>bmc.pch.load</td>
<td>Library that contains your product code</td>
<td>Valid data set name</td>
</tr>
<tr>
<td>bmc.uim.content</td>
<td>Library that contains code and files that are downloaded to a console during console installation</td>
<td>Valid data set name</td>
</tr>
<tr>
<td>HLQ.customized.config</td>
<td>UIM Customized Config Library, as specified in the Installation System This library will contain customized configuration members.</td>
<td>Valid data set name</td>
</tr>
<tr>
<td>bmc.uim.config</td>
<td>Library that contains common UIM server, console code, and product execution parameters that are used during UIM server initialization</td>
<td>Valid data set name</td>
</tr>
</tbody>
</table>

4 Save the new #UIMX member.

Defining the MainView Infrastructure CAS connection (optional)

You have the option of using the UIM server to pass threshold data that is pushed from MainView Threshold Advisor.
If you plan to do that, use the following procedure to define a MainView Infrastructure coordinating address space (CAS) connection.

**To define a MainView Infrastructure CAS connection**

1. Locate the UIMM$PCH member in the UIM sample library.
2. Copy the UIMM$PCH member to your HLQ.customized.config library.

    Figure 74 on page 240 shows the UIMM$PCH member.

    **Figure 74: Default UIMM$PCH member**

    ```
    <BMC_CHTTP>
      <!--
      *** Assembler PCH for MVS and Windows -
      Skeleton for INITPROCESS / TERMPROCESS ***
      -->

      <BMC_PRODUCT_FAMILY ID="MVI-PAS-STARTUP-ASSIST">
        <PRODUCT_CODE_HANDLER ID="UIM0UIPH"
          MODULE="UIMMV00"
          AFFINITY="ONLY"/>
          <!--
          The Current BMC_PARM values passed are those needed by
          BBM9XZZO (MVI Service Point Attach.) Others will be added
          as new functionality is added.
          -->

        <BMC_PARM ID="SSID"
          VALUE="BBCS"/>    <!-- CAS to connect to -->
        <BMC_PARM ID="XDM"
          VALUE="NO"/>      <!-- YES|NO -->
      </BMCPRODUCT_FAMILY>
    </BMC_CHTTP>
    ```

3. In the copied member, change the ID="SSID" parameter from BBCS to the subsystem ID of the MainView Infrastructure CAS with which you want UIM to communicate.

    The CAS that you identify receives threshold data that is pushed from MainView Threshold Advisor.

4. Save the updated UIMM$PCH member.

**Setting up the HFS data set**

After you create the startup configuration member and the started task procedure, you can allocate and, optionally, initialize the HFS data set.

This server-side storage data set stores user preferences and dynamic configuration information on the UIM server. During installation, the HLQ.HFS member in the
sample library was created and customized with your site-specific information. You can submit the customized member as is, or you can customize a copy of the member in the sample library.

You can share the HFS data set between all UIM servers within a sysplex or all LPARs that share the DASD complex. Alternatively, you can create a unique HFS data set for each UIM server.

## Allocating the HFS data set

Use the following procedure to allocate the HFS data set by customizing a copy of the `HLQ.HFS` member in the UIM sample library.

### To customize a copy of the data set member

1. Locate the `#DEFHFS` member in the UIM sample library.

   Figure 75 on page 241 shows the `#DEFHFS` member. The `HLQ` value is the high-level qualifier that you specified during installation.

   **Figure 75: Default #DEFHFS member**
   ```
   //ALLPDSE EXEC PGM=IEFBR14
   //HFSPDSE DD DISP=(NEW,CATLG),UNIT=SYSDA,SPACE=(CYL,(1,1)),
   //    DCB=(DSORG=PO,RECFM=VB,LRECL=4096),
   //    DSNTYPE=LIBRARY,
   //    DSN=HLQ.HFS
   ```

2. Edit the `#DEFHFS` member by changing the variables listed in Table 25 on page 241.

   **Table 25: #DEFHFS member variables**
<table>
<thead>
<tr>
<th>Variable</th>
<th>Definition</th>
<th>Accepted value</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>Device for placing data sets</td>
<td>See your site standards.</td>
</tr>
<tr>
<td>DSN</td>
<td>Data set name</td>
<td>See your site standards.</td>
</tr>
</tbody>
</table>

3. Save the edited `#DEFHFS` member with a new name.

4. Submit the DEFHFSJC JCL member.

### Initializing the HFS data set manually *(optional)*

By default, the HFS data set that you allocated is initialized when the UIM server is started.

If you need to initialize the data set *manually*, use the following procedure.
To initialize the HFS data set manually

1 Locate #LOADHFS in the UIM sample library.

Figure 76 on page 242 shows the default #LOADHFS member.

Figure 76: Default #LOADHFS member

```*
//******************************
//* Load the BMC HFS PDSE with SAMP library members.
//* Optionally convert previous BMC HFS HostList.xml if found.
//******************************

LOADHFS EXEC PGM=UIMHFSL,REGION=0K,
// PARM='=version $UIMHFSL $UIMHFST'
/* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
/* UIMHFSL parameters:
/*
/* memname  Samplib control cards member, default is $HFSLOAD.
/*  - $HFSLOAD is a Load of default properties files
/*  -t      Activate tracing to SYSPRINT
/*
/* SAS/C Runtime Library parameters:
/*
/* =version  Print Runtime Library release information to SYSTERM
/*
/*-------------------------------
/STEPLIB DD DISP=SHR,DSN=???????????.LOAD
/*
/* SAS/C DD'S
/*SYSTERM DD SYSOUT=* 
/*SYSPRINT DD SYSOUT=* 
/*STGRPT DD SYSOUT=* 
/*
/* STANDARD JOB DD'S
/*SYSDUMP DD SYSOUT=* 
/*
/*HFSPDSE DD DISP=SHR,DSN=???????????.HFS
/*SAMP DD DISP=SHR,DSN=???????????.SAMP
/*CONT DD DISP=SHR,DSN=???????????.CONTENT
/*
```

2 Edit the #LOADHFS member, and change ????????? to a high-level qualifier that meets your site standards.

3 Save the edited member as HFSLOAD.

Note
This step overwrites the UIM Customized Parm Library member that was created during installation.

4 Submit the HFSLOAD member JCL.
Configuring the UIM server

The UIM server is customized during installation.

However, you can change the original configuration of the UIM server permanently by changing the values for parameters in the startup and trace members of the configuration file.

You can change the following UIM server configuration options:

- Port number
- Security authorization timeout
- Idle timeout for affinity tasks
- HFS server-side storage data set
- Enable/disable tracing
- Enable/disable network browser command interface
- Enable/disable encrypted credentials
- USS HFS root directory
- MainView CAS connection (*for MainView products*)

Each UIM server requires a configuration member that describes the unique characteristics of the server. This member is specified as a parameter in the UIM server configuration file.

The UIM sample library (*HLQSAMP*) contains a template for the configuration member #NORMAL. The installation process customizes the #NORMAL member, gives it the same name as the started task procedure, and copies the member to the UIM Customized Config Library.

*Figure 77 on page 243* shows the #NORMAL member.

**Figure 77: #NORMAL member template**

```
<BMCHTTP>
  <BMC_PARM ID="PORT" VALUE="3683" />
  <BMC_PARM ID="AUTH_TIMEOUT_SECS" VALUE="1800" />
  <BMC_PARM ID="AFF_TIMEOUT_SECS" VALUE="1800" />
  <BMC_PARM ID="HFS_DATASET" VALUE="HLQ.HFS" />
  <BMC_PARM ID="ALLOW_NETCMD" VALUE="YES" />
```
Table 26 on page 244 lists the parameters in the #NORMAL member.

**Table 26: #NORMAL member parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
<th>Accepted values</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;PORT&quot; VALUE=&quot;3683&quot;/&gt;</code></td>
<td>Port value for the UIM server</td>
<td>Unique numeric value that is between 1 and 65535 The default is 3683.</td>
<td>“Changing the port number” on page 245</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;AUTH_TIMEOUT_SECS&quot; VALUE=&quot;1800&quot; /&gt;</code></td>
<td>Timeout for security authorization</td>
<td>Numeric value in seconds The default is 1800.</td>
<td>“Changing the security authorization timeout feature permanently” on page 245</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;AFF_TIMEOUT_SECS&quot; VALUE=&quot;1800&quot; /&gt;</code></td>
<td>Idle timeout period for affinity tasks</td>
<td>Unique numeric value in seconds The default is 1800.</td>
<td>“Changing the idle timeout for affinity tasks” on page 246</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;HFS_DATASET&quot; VALUE=&quot;HLQ.HFS&quot; /&gt;</code></td>
<td>HFS data set name</td>
<td>PDSE data set name for storing HFS data</td>
<td>“Changing the HFS server-side storage data set name” on page 246</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;ALLOW_NETCMD&quot; VALUE=&quot;YES&quot; /&gt;</code></td>
<td>Whether to enable the network browser command interface</td>
<td>YES (default), NO, or AUTHORIZE</td>
<td>“Enabling or disabling the network browser command interface” on page 247</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;ENCRYPTION_LEVEL&quot; VALUE=&quot;CREDENTIALS-IF&quot; /&gt;</code></td>
<td>Whether to enable encryption of user credentials</td>
<td>CREDENTIALS-IF (default) or NO</td>
<td>“Enabling or disabling encryption of user credentials” on page 247</td>
</tr>
<tr>
<td><code>&lt;BMC_PARM ID=&quot;USS_HFS_ROOT&quot; VALUE=&quot;/tmp/bmcuim&quot; /&gt;</code></td>
<td>Root directory in USS HFS for web service configuration members</td>
<td>/tmp/bmcuim (default)</td>
<td>“Setting the USS HFS root directory” on page 248</td>
</tr>
</tbody>
</table>
Changing the port number

The port number for the UIM server is the address of the server’s TCP/IP application on an IBM z/OS image.

The UIM server has one registered port number that consoles use to contact the server.

Use the following procedure to change the port number globally for all consoles that communicate with the UIM server.

To change the port number

1. In your startup configuration member, find the PORT parameter:

   ```xml
   <BMC_PARM ID="PORT" VALUE="3683" />
   ```

   The default registered port number is 3683.

2. Change the value to a unique port number between 1 and 65535.

   **WARNING**

   Check with your TCP/IP administrator to ensure that you are entering a unique port number. If you do not enter a unique port number, program errors might occur.

Changing the security authorization timeout feature permanently

A console is equipped with a timeout security feature.

This feature controls the amount of time that all consoles that communicate with the UIM server can remain inactive before security authorization expires. This value is set during installation.

Use the following procedure to change the timeout feature permanently for all consoles that communicate with the UIM server.

**Note**

To change the timeout feature temporarily, see “Changing the security authorization timeout feature temporarily” on page 255.
To change the timeout feature permanently

1 In your startup configuration member, find the AUTH_TIMEOUT_SECS parameter:

```xml
<BMC_PARM ID="AUTH_TIMEOUT_SECS"
VALUE="1800" />
```

The default number of seconds is 1800.

2 Change the value to any number of seconds.

Changing the idle timeout for affinity tasks

Affinity timeout is the amount of time that the task is held or idle between requests for the affinity.

When the task is idle for the defined period, the affinity is no longer valid, and the task is available for other work.

Use the following procedure to change the idle timeout value.

To change the idle timeout for affinity tasks

1 In your startup configuration member, find the AFF_TIMEOUT_SECS parameter:

```xml
<BMC_PARM ID="AFF_TIMEOUT_SECS"
VALUE="1800" />
```

The default number of seconds is 1800.

2 Change the value to any number of seconds.

Changing the HFS server-side storage data set name

The HFS server-side storage data set stores user preferences and dynamic configuration information on the UIM server.

Use the following procedure to change the HFS data set name.

To change the HFS data set name

1 In your startup configuration member, find the HFS_DATASET parameter:

```xml
<BMC_PARM ID="HFS_DATASET"
VALUE="HLQ.HFS" />
```

This parameter specifies the data set name to use for the UIM server server-side storage data set.
2 Change the value of the high-level qualifier for HFS_DATASET from HLQ to a value that meets your site standards.

**Enabling or disabling the network browser command interface**

The network browser command interface displays UIM server information and allows an administrator to make dynamic modifications to UIM server settings.

This interface is also known as the BMC UIM Server Commands web page.

Use the following procedure to enable or disable the network command interface. You can also provide an active authentication with the browse session. After the variable ALLOW_NETCMD is set to AUTH, the user must log in by using the following command in the web browser:

```
http://uimServerHostName:uimPortNumber/UIMLogon
```

If the logon is successful, the user can display the Commands web page (http://uimServerHostName:uimPortNumber/htpcmd.html). For more information about the Commands web page, see “Accessing the BMC UIM Server Commands web page” on page 253.

**To enable or disable the network browser command interface**

1 In your startup configuration member, find the ALLOW_NETCMD parameter:

```
<BMC_PARM ID="ALLOW_NETCMD" VALUE="YES" />
```

By default, the UIM server processes network browser commands, but you can disable those commands by setting ALLOW_NETCMD to NO.

2 Change ALLOW_NETCMD to one of the following values:

- YES to enable the network browsing command interface
- NO to disable the network browsing command interface
- AUTH to require logging on via the UIM logon URL

**Enabling or disabling encryption of user credentials**

If your IBM zSeries processor supports HMAC-SHA and 3DES, UIM can use them to encrypt a user's ID and password.

Use the following procedure to change the encryption option.
To enable or disable encryption of user credentials

1 In your startup configuration member, find the ENCRYPTION_LEVEL parameter:

```xml
<BMC_PARM ID="ENCRYPTION_LEVEL" VALUE="CREDENTIALS-IF" />
```

The default is to encrypt user credentials if the client supports encryption (CREDENTIALS-IF).

2 Change ENCRYPTION_LEVEL to one of the following values:

- CREDENTIALS - IF to encrypt user credentials if the client supports encryption
- NO to disable encryption of user credentials

---

**WARNING**

If you set ENCRYPTION_LEVEL to NO, user credentials are transmitted in clear text.

---

Setting the USS HFS root directory

UIM web services require a directory path in USS for storing configuration members. The UIM server must be able to read and write to this path. Use the following procedure to set the USS HFS root directory.

To set the USS HFS root directory

1 In your startup configuration member, find the USS_HFS_ROOT parameter:

```xml
<BMC_PARM ID="USS_HFS_ROOT" VALUE="/tmp/bmcuim" />
```

The default directory path is /tmp/bmcuim, but you can specify a different directory.

2 Change the value to a valid directory path in USS.

---

**Note**

The UIM started task user account must have read/write access to the specified USS HFS root directory.
Enabling or disabling the overall tracing option permanently

The UIM server is installed with the recommended tracing options preset.

Each server requires a trace configuration member that defines the tracing for that server. The UIM sample library contains a trace configuration member called TRACE. During installation, the TRACE member (Figure 78 on page 249) is copied to the UIM Customized Config Library.

**Note**

BMC recommends that you change the trace configuration member (permanently or temporarily) only when Customer Support directs you to do so.

To change the overall tracing option temporarily, see “Enabling or disabling the overall tracing option temporarily” on page 256.

**Figure 78: TRACE member**

```xml
<BMCHTTP>
  <RRLOG VALUE="OFF"/>
  <TRACE VALUE="ON">
    <BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_WARNING"/>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_INFO" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_ENTRYEXIT" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_Control" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_MEMORY" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_SOCKET" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_THREAD" ->>
    <!-.- BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_WAIT" ->>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_SERVER"/>
    <!-.- BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_SERVER_WL" ->>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_CLIENT"/>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_REQUEST"/>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_EXTENSION"/>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_TASKMGR"/>
    <!-.- BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_TASKWKR" ->>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_LOGTASK" ->>
    <BMC_PARM ID="TRACE_COMPONENT" VALUE="TRACECOMPONENT_EXTERNAL"/>
  </TRACE>
</BMCHTHTTP>
```
To change the overall tracing option permanently

1. In your trace configuration member, find the TRACE parameter.

2. Change TRACE to one of the following values:
   - ON to enable the overall tracing option
   - OFF to disable the overall tracing option

3. Verify that tracing has been enabled or disabled.

Enabling or disabling specific tracing options permanently

The UIM server is installed with recommended specific tracing options preset.

Use the following procedure to change specific tracing options permanently.

**Note**

BMC recommends that you change the trace configuration member only when Customer Support directs you to do so.

To change specific tracing options temporarily, see “Enabling or disabling specific tracing options temporarily” on page 256.

To verify that the overall tracing option is enabled

1. In your trace configuration member, verify that the overall trace option (TRACE) is enabled:

   ```xml
   <TRACE VALUE="ON">
   ```

2. If the TRACE parameter is set to OFF, change its value to ON and save the trace configuration member.

To enable specific tracing options

1. From the list of specific tracing options in your trace configuration member, find the option that you want to enable.

   The following example shows the TRACE_ACTION option:

   ```xml
   <!-- BMC_PARM ID="TRACE_ACTION"
   VALUE="TRACEACTION_INFO" -->
   ```

2. Remove the beginning and ending characters that identify the option as a comment:
After the opening bracket (<), delete !-- and the spaces immediately following.

Before the closing bracket (>), delete the two hyphens (--).

The example from Step 1 on page 250 would appear as follows. With the comment characters removed, the option is now enabled.

```xml
<BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_INFO" />
```

To disable specific tracing options

1. From the list of specific tracing options in your trace configuration member, find the option that you want to disable.

   The following example shows the TRACE_ACTION option:

   ```xml
   <BMC_PARM ID="TRACE_ACTION" VALUE="TRACEACTION_INFO" />
   ```

2. Add the beginning and ending characters that identify the option as a comment:
   - After the opening bracket (<), add !-- and a space.
   - Before the closing bracket (>), add two hyphens (--).

   The example from Step 1 on page 251 would appear as follows. With the comment characters added, the option is now disabled.

   ```xml
   <!-- BMC_PARM    ID="TRACE_ACTION" VALUE="TRACEACTION_INFO" -->
   ```

Changing the MainView CAS connection

Use the following procedure to change the MainView CAS to which the UIM server connects.

To change the MainView CAS connection

1. Find the UIMM$PCH member in your customized configuration library.

2. Change the ID="SSID" parameter to the subsystem ID of the MainView Infrastructure CAS with which you want UIM to communicate.

   The CAS that you identify receives threshold data that is pushed from MainView Threshold Advisor.
Operating the UIM server

To control the operation of the UIM server, you can perform the following tasks:

- From the MVS operator console or a comparable operator facility
  - Start and stop the server
  - Verify that the server is running
- From a web browser
  - View active users
  - Change the security authorization feature
  - Enable or disable tracing options temporarily
  - Refresh the UIM server content

Starting and stopping the UIM server

The UIM server must be running before you attempt to connect to a console.

If the UIM server is not running, you cannot:

- Run or install a console
- Access any UIM commands through a web browser

To start and stop the UIM server, you must issue MVS operator commands on the host computer where the UIM server is installed.

WARNING
To avoid data loss, notify active users when you must stop the UIM server.

To start the UIM server

1. Issue the following MVS operator command:

   \$ uimServerName

   uimServerName is the name of the UIM server started task.
To stop the UIM server

1. Issue the following MVS operator command:

   `P uimServerName`

Verifying that the UIM server is running

Use the following procedure to verify whether the UIM server is running by reviewing the JESMSGLG SYSOUT file.

To verify that the UIM server is running

1. Review the JESMSGLG SYSOUT file for the following messages:

   - BMC340290I UIM server, Level V.R.MM MM, DD,YY, initialization complete!
   - BMC340122I Ready for MVS Operator Commands

Accessing the BMC UIM Server Commands web page

All UIM server operation tasks are performed from the BMC UIM Server Commands web page. The UIM server must be started before you can access this web page.

Authorization settings

Before you can access the BMC UIM Server Commands web page, you must set the ALLOW_NETCMD parameter in the UIM startup member. Table 27 on page 253 lists the available ALLOW_NETCMD settings.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>The Commands web page is disabled.</td>
</tr>
<tr>
<td>YES</td>
<td>The Commands web page is enabled.</td>
</tr>
<tr>
<td>AUTH</td>
<td>You must log into the UIM server Logon web page to verify that you are allowed to access the Commands web page.</td>
</tr>
</tbody>
</table>

Table 27: ALLOW_NETCMD variable settings

If AUTH is set in the UIM startup member, and you are not authorized to access the Commands web page, the following message is displayed in your browser when you try to access the web page:

403 - Access to URN not allowed
If you see this message, complete the procedure “To access the BMC UIM Server Commands web page using authorization” on page 254.

To access the BMC UIM Server Commands web page

1 From a supported web browser, enter the following URL for the BMC UIM Server Commands web page:

   http://uimServerHostName:uimPortNumber/htpcmd.html

   The variables in the URL are defined as follows:

   - **uimServerHostName** is the name of the host computer on which the UIM server is installed.
   - **uimPortNumber** is the port number that is assigned to the UIM server. The default is 3683.

   For example, if the host is SYSO and the port number is 3683, enter http://syso:3683/htpcmd.html.

To access the BMC UIM Server Commands web page using authorization

1 From a supported web browser, enter the following URL for a console UIM server Logon web page:

   http://uimServerHostName:uimPortNumber/UIMLogon

2 Enter your user ID and password.

3 (optional) Enter a group identification and account.

   **Note**

   You can change the password on the UIM host by entering a new password in the New Password field.

4 Click OK.

   The message Logon successful is displayed in your web browser. The authentication for the UIM server is stored in a cookie in your web browser.

5 Enter the following URL for the BMC UIM Server Commands web page:

   http://uimServerHostName:uimPortNumber/htpcmd.html

6 To stop access to the Commands web page, enter the following URL:

   http://uimServerHostName:uimPortNumber/UIMLogoff
Viewing active users

Use the following procedure to view the names of active users (any users who are logged on to a console).

To view the names of active users

1. Access the BMC UIM Server Commands web page as described in “Accessing the BMC UIM Server Commands web page” on page 253.

2. Click Display Active Users.

   The UIM server Active User Display page lists the names of active users.

3. Click Back to UIM Server Commands to return to the Commands web page.

Changing the security authorization timeout feature temporarily

Use the following procedure to change the security authorization timeout feature temporarily for all consoles that communicate with the UIM server.

The change remains in effect until the UIM server is stopped.

To change the timeout feature temporarily

1. Access the BMC UIM Server Commands web page as described in “Accessing the BMC UIM Server Commands web page” on page 253.

2. Click Display Active Users.

3. In the Inactivity Time Out box, type a value (in minutes or seconds) that represents the amount of time a console can remain inactive without timing out.

4. Choose a unit of time for the console timeout feature by selecting Seconds or Minutes.

5. Click Change to apply your changes.

6. Click Back to UIM Server Commands to return to the Commands web page.
Enabling or disabling the overall tracing option temporarily

Use the following procedure to enable or disable the overall tracing option for the UIM server temporarily.

The change remains in effect until the UIM server is stopped or until you change the option again.

Trace data is written to the TRCLOG DD. To view trace data, review the contents of the TRCLOG DD.

To enable or disable the overall tracing option temporarily

1. Access the BMC UIM Server Commands web page as described in “Accessing the BMC UIM Server Commands web page” on page 253.

2. Click Internal Trace - Display Trace Status/Modification.

3. In the Overall Trace column, click either Active or Inactive.

   - If the value is Inactive when you click on it, the value changes to Active and the overall tracing option is enabled for the UIM server.
   
   - If the value is Active when you click on it, the value changes to Inactive and the overall tracing option is disabled for the UIM server.

4. Click Back to UIM Server Commands to return to the Commands web page.

Enabling or disabling specific tracing options temporarily

Use the following procedure to enable or disable specific tracing options for the UIM server temporarily.

The change remains in effect until the UIM server is stopped or until you change the option again.

Trace data is written to the TRCLOG DD. To view trace data, review the contents of the TRCLOG DD.

To enable or disable specific tracing options temporarily

1. Access the BMC UIM Server Commands web page as described in “Accessing the BMC UIM Server Commands web page” on page 253.
2 Click **Internal Trace - Display Trace Status/Modification**.

3 In the **Overall Trace Indicators** section, ensure that the **Overall Trace** option is set to **Active**.

   **Note**
   The overall trace option must be enabled before you can enable or disable specific tracing options.

4 In the **Trace Components Indicators** section, find the category of tracing options that you want to enable or disable:

   - Trace components
   - Trace actions

5 To enable or disable specific tracing options, click either **On** or **Off** in the **Status** column.

   The **Status** column works as a toggle:

   - If the value is **Off** when you click on it, the value changes to **On** and that specific tracing option is enabled.
   - If the value is **On** when you click on it, the value changes to **Off** and that specific tracing option is disabled.

6 Click **Back to UIM Server Commands** to return to the Commands web page.

### Refreshing the UIM server content

Use the following procedure to refresh the content of the UIM server without shutting it down.

   **Note**
   You might need to refresh the content of the server when a new version is installed. However, this task is not usually required.

#### To refresh UIM server content

1 Access the BMC UIM Server Commands web page as described in “Accessing the BMC UIM Server Commands web page” on page 253.

2 Click **Display/Refresh Contents Directory**.

3 On the MVS Content Index page, click **Refresh Content Index**.
Troubleshooting UIM server problems

This section discusses specific UIM server-related problems that you might encounter.

**Note**
If you contact Customer Support, they might ask you to enable or disable overall tracing, or specific tracing options. The following topics discuss enabling and disabling tracing options:

- “Enabling or disabling the overall tracing option temporarily” on page 256
- “Enabling or disabling specific tracing options temporarily” on page 256

**UIM server does not start because the OMVS segment is not defined**

If the started task is missing an IBM RACF OMVS segment, one of the following messages is displayed in the job log output when the UIM server start command is issued:

- Message IEF695I, stating that the job name is assigned to the specified user
- Message ICH408I, stating that the OMVS segment is not defined

In addition, message LSCX902 is displayed in the SYSTERM DD output and states that an MVS initialization error occurred.

To solve this problem, include a RACF OMVS segment for the UIM server address space.

**To determine whether a started task is missing a RACF OMVS segment**

1. Check the JES message log for the following messages:

   - IEF695I START RGSTEMP WITH JOBNAME RGSTEMP IS ASSIGNED TO USER ++++++++ 
   - ICH408I JOB(RGSTEMP ) STEP(RGSTEMP ) CL(PROCESS ) OMVS SEGMENT NOT DEFINED
2. Check the SYSTERM DD output for the following message:

```
LSCX902 **** WARNING **** ERRNO = EMVSINITIAL
Generated in PFSCTL called from line ...
```

If you find this message, one of the following conditions exists:

- No RACF rule has assigned a user to the started task.
- A RACF rule has assigned a user to the started task, but the user does not have an associated OMVS segment.

The UIM server uses TCP/IP. TCP/IP requires UNIX system services, and a RACF OMVS segment must exist for the UIM server address space. To define an OMVS segment for a user of the started tasks, contact your security administrator.

**UIM server does not start and displays message LSCX902 in SYSTERM DD**

If the TCP/IP started task is not named TCPIP, message LSCX902 is displayed in the SYSTERM DD output and the UIM server does not start.

To solve this problem, change the name of the TCP/IP started task name to TCPIP. The root cause of the problem is that the UIM server cannot find the TCP/IP address space due to one of the following conditions:

- No TCP/IP address space with the started task name TCPIP exists because the address space has been given another name.
- Multiple TCP/IP address spaces exist, but none are named TCPIP.
- Multiple TCP/IP address spaces exist, but you do not want to use the started task named TCPIP.

To change the name of the TCP/IP started task name to TCPIP, perform one of the following tasks:

- Determine the name of the TCPIP file (TCPIP.DATA). TCP/IP client address spaces use this file to determine local TCP/IP configuration information.
  
  The local TCP/IP configuration information includes the name of the TCP/IP address space that the TCPIPUSERID parameter specifies. The UIM server procedure must be updated to include a SYSTCPD DD statement that specifies the appropriate TCPIP.DATA file.

- Specify the TCP/IP address space name in a parameter to the UIM server by using the procedure parameter ENV, as follows:

  ```
  // ENV="=TCPIP_MACH=startedTaskName"
  ```
Message LSCX902 is displayed in SYSTERM DD

If Interlink or CA TCPAccess is installed instead of IBM TCP/IP Stack, message LSCX902 is displayed in the SYSTERM DD output.

To solve this problem, insert the TCPAccess LINKLIB as the first data set in the UIM server STEPLIB DD. The TCPAccess LINKLIB must be the first data set in the STEPLIB concatenation. Inserting the LINKLIB first provides the correct socket API interface modules.

**WARNING**

Insert only the LINKLIB library, not the LOAD library, because TCPAccess uses SAS/C. Using TCPAccess SAS/C might cause an incompatibility in the run-time modules.

Message LSCX904 is displayed in SYSTERM DD

If the TCPIP PROFILE file specifies a reserved port number for a started task name that does not match the UIM server started task name, message LSCX904 is displayed in the SYSTERM DD output.

To solve this problem, specify another port number for the UIM server, or correct the started task name.

To specify a different port number in the TCIP PROFILE file

1. Review the SYSTERM DD output for the following message:

   LSCX904 **** WARNING **** ERRNO = EACCES
   Generated in BIND called from line
   Unexpected failure in bind, reason code 744C7246

   If the message is in the SYSTEM DD output, the wrong name job name was used in the TCPIP PROFILE file.

2. Select another port number for the UIM server.

   The following is a sample entry in the TCPIP PROFILE file:

   8300 TCP STFTUIM ; BMC UIM server

   This entry reserves the port number 8300 for the job named STFTUIM that is associated with the UIM server. Any other job trying to access that port number receives message LSCX904 in the SYSTERM DD output.
## Controlling diagnostic messages

### Tip
The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

This chapter explains how to use the DIAGMSG view to set monitoring options that regulate the type and extent of diagnostic messages produced by MainView when operating in windows mode.

The Plex Manager DIAGMSG view displays the current state of MainView diagnostic monitoring. Using the DIAGMSG view line commands, options can be set that regulate the type of messages produced by MainView components and an external security manager (ESM) such as IBM RACF or CA ACF2.

### Displaying the DIAGMSG view

The following figure shows an example of the DIAGMSG view.

For more information about any field in the DIAGMSG view, place the cursor on the field and press the Help key.

**Figure 79: DIAGMSG view**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CURRENT WIN</th>
<th>ALT WIN</th>
<th>W1 - DIAGMSG View</th>
<th>SYSB</th>
<th>ddmmmyyyy</th>
<th>hh:mm:ss</th>
<th>PLEXMGR</th>
<th>D</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>GXDM</td>
<td>OFF</td>
<td>Global</td>
<td>Extended Diagnostic Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LXDM</td>
<td>OFF</td>
<td>Local</td>
<td>Extended Diagnostic Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEMM</td>
<td>ON</td>
<td>Global</td>
<td>Extended Message Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LEMM</td>
<td>OFF</td>
<td>Local</td>
<td>Extended Message Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSEMM</td>
<td>OFF</td>
<td>Local</td>
<td>Security Extended Message Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LESTR</td>
<td>OFF</td>
<td>Local</td>
<td>Extended Security Trace</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GESTR</td>
<td>OFF</td>
<td>Global</td>
<td>Extended Security Trace</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSSTR</td>
<td>OFF</td>
<td>Local</td>
<td>Simple Security Trace</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSSTR</td>
<td>OFF</td>
<td>Global</td>
<td>Simple Security Trace</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSSM</td>
<td>OFF</td>
<td>Global</td>
<td>Safe Security Message Display</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAFTRACE</td>
<td>N/A</td>
<td>acf2/mvs</td>
<td>ACF2 SAFTRACE GSO Option</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAFTRC</td>
<td>N/A</td>
<td>LogonID</td>
<td>ACF2 LogonID SAF-TRC attribute</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WSXASTR</td>
<td>OFF</td>
<td>Window</td>
<td>Extended Authorization Simple Trace</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
To access the DIAGMSG view

1. Perform one of the following actions:
   - In Plex Manager, type DIAGMSG on the COMMAND line and press Enter.
   - From any MainView product that is running in windows mode, proceed as follows:
     1. From the MainView Selection Menu, select option P, Plex Manager.
        The EZPLEX menu is displayed.
     2. On the EZPLEX menu under Diagnostics, position the cursor on Messages and press Enter.
   - Alternatively, use the following CONtext command to access Plex Manager and display the DIAGMSG view:
     CON * PLEXMGR;DIAGMSG
     The abbreviated name of each diagnostic message or debugging option is shown in the Option field of the DIAGMSG view. The Status field shows the current state (On or Off) of the option. The Scope field indicates the extent of the MainView environment that is being monitored by a specific diagnostic option.

     In the CMD field next to the DIAGMSG diagnostic option whose state you want to change, enter one of the following line commands:
     - ON to enable the diagnostic option
     - OFF to disable the diagnostic option

Understanding message and diagnostic options

Each option that is listed in the DIAGMSG view represents a MainView diagnostic mode with a characteristic scope and a set of associated messages that can be issued if the mode is active.

Valid MainView diagnostic or message modes are described in Table 28 on page 263.
<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| GXDM   | Global Extended Diagnostic Mode  
Global Extended Diagnostic Mode extends Extended Diagnostic Mode (XDM) monitoring to the entire MainView subsystem, which includes the CAS and all connected product and user address spaces.  
GXDM mode is set by the XDM=Y parameter of the CAS started task procedure.  
Extended Diagnostic Mode initiates additional internal diagnostic and validation procedures. It is not recommended for normal MainView operation. In addition, Extended Diagnostic Mode causes all Extended Message Mode messages to be issued in the same manner as the GXDM option. |
| LXDM   | Local Extended Diagnostic Mode  
Local Extended Diagnostic Mode sets monitoring conditions for the current CAS, PAS, or UAS. Local Extended Diagnostic Mode issues all Extended Message Mode messages similar to the LEMM option.  
Local Extended Diagnostic Mode initiates additional internal diagnostic and validation activity. It is not recommended for normal MainView operation. |
| GEMM   | Global Extended Message Mode  
Global Extended Message Mode sets monitoring conditions for diagnostic messages issued by the CAS and any connected address space, such as a PAS or a UAS.  
Global Extended Message Mode issues standard diagnostic messages similar to the GXDM option. |
| LEMM   | Local Extended Message Mode  
Local Extended Message Mode sets monitoring conditions for diagnostic messages issued by the current address space.  
Standard security diagnostic messages that are controlled by LSEMM mode are a subset of LEMM messages. Therefore, LSEMM messages are issued when LEMM mode is active, regardless of LSEMM status.  
Likewise, when LEMM mode is active, you cannot disable standard security messages by disabling the LSEMM option. |
| LSEMM  | Local Security Extended Message Mode  
Local Security Extended Message Mode sets monitoring conditions for standard security diagnostic messages that are issued by the current address space.  
The messages that are controlled by LSEMM mode are a subset of the messages managed by LEMM mode. LSEMM mode can manage the set of security-oriented messages independently of LEMM mode. However, if LEMM mode is active, you cannot prevent security diagnostic messages from being issued by inactivating LSEMM mode. |
<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| LESTR  | Local Extended Security Trace  
Local Extended Security Trace mode manages extended diagnostic security trace messages that are produced by the current address space.  
These messages are in addition to LSEMM messages and provide more information than the standard security trace messages.  
Local Simple Security Trace messages (LSSTR) are a subset of the security trace messages that are managed by LESTR mode. |
| GESTR  | Global Extended Security Trace  
Global Extended Security Trace mode manages extended diagnostic security trace messages that are produced by the CAS and all connected address spaces.  
These messages are in addition to the LSEMM option messages and provide more information than standard security trace messages.  
The messages that are managed by GSSTR mode are a subset of GESTR messages. When GESTR mode is active, GSSTR messages are displayed regardless of the state of the GSSTR option. |
| LSSTR  | Local Simple Security Trace  
Local Simple Security Trace mode manages simple diagnostic security trace messages that are issued by the current address space.  
LSSTR messages are a subset of the messages that are managed by the LESTR option.  
When LESTR mode is active, LSSTR messages are displayed regardless of the state of the LSSTR option. |
| GSSTR  | Global Simple Security Trace  
Global Simple Security Trace mode manages simple diagnostic security trace messages that are issued by the CAS and all connected address spaces.  
GSSTR mode allows simple security trace messages to be managed independently of GESTR mode messages. |
| GSSM   | Global Safe Security Message  
Global Safe Security Message mode manages safe security messages that are issued by an external security manager during the initialization phase of a MainView window.  
The default value is OFF; safe security messages are not displayed.  
Safe security messages are standard messages that are issued during the creation of a security environment by the external security manager at sign-on. Safe security messages include message IDs, such as BBMSS201I, ICH70001I, TSS7000I, TSS7001I, and ACF01137, plus any additional messages designated by your installation.  
Inherited security environments that are established during window initialization produce messages that normally convey insignificant information. These messages are typically bypassed. In cases where these messages might be important, the GSSM option can be set ON to display these messages under all circumstances.  
If the external security manager returns any messages other than safe security messages during window initialization, this option value is ignored and all security messages are displayed. |
<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| WSXASTR| Extended Authorization Simple Trace  
Extended Authorization Simple Trace mode manages simple Extended Authorization security interface trace messages that are issued in response to target or context activity occurring in the current window.  
Extended Authorization is the name of the internal mechanism that is used by most components to authorize end user access to resources protected by the security interface. The simple trace messages issued by Extended Authorization are more general than those issued as a result of activating the LSSTR option.  
You can dynamically enable or disable the display of simple trace messages for Extended Authorization security calls from the current window by specifying the appropriate line command.  
A value of **OFF** means that Extended Authorization trace messages are disabled for the current window and are not displayed.  
A value of **ON** means that Extended Authorization trace messages are enabled for the current window and are displayed in a manner similar to error messages. |
| SAFTRACE| CA ACF2 SAFTRACE GSO Option Status  
CA ACF2 SAFTRACE GSO Option Status mode manages messages that are produced by the CA ACF2 external security manager in the current context.  
The SAFTRACE option indicates the status of the CA ACF2 5.2 SAFTRACE GSO option, which is maintained by CA ACF2 for each system image. You can alter the status of this option only with facilities that are provided by CA ACF2 for this purpose.  
The SAFTRACE option status is shown because it must be enabled to control SAFTRC messages.  
A value of **ON** in the Status column indicates the CA ACF2 5.2 SAFTRACE GSO option is enabled. Requests for SAFTRC messages are supported.  
A value of **OFF** in the Status column indicates the CA ACF2 5.2 SAFTRACE GSO option is disabled and the CA ACF2 NOSAFTRACE GSO option is in effect. Requests for SAFTRC messages are ignored.  
For CA ACF2 Release 6.0 and above, the SAFTRACE GSO option is not supported and is always shown as **OFF**. |
Understanding the scope of an option

The **Scope** field on the DIAGMSG view indicates the extent that an option monitors activity across a system image.

Valid scope values are shown in the following table:

<table>
<thead>
<tr>
<th>Scope</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA ACF2/MVS</td>
<td>Diagnostic monitoring occurs over the entire range of a CA ACF2 security subsystem running on the current system image. The CA ACF2/MVS scope requires the LSEMM option to be enabled or CA ACF2 to be the designated external security manager in the current context.</td>
</tr>
<tr>
<td>Global</td>
<td>Diagnostic monitoring applies to the CAS and all connected address spaces.</td>
</tr>
<tr>
<td>Local</td>
<td>Diagnostic monitoring applies to the current address space only.</td>
</tr>
<tr>
<td>LogonID</td>
<td>Diagnostic monitoring is limited to the current CA ACF2 LOGONID address space. The LogonID scope requires the LSEMM option be enabled or CA ACF2 be the external security manager in the current context.</td>
</tr>
</tbody>
</table>
### Understanding the scope of an option

<table>
<thead>
<tr>
<th>Scope</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window</td>
<td>Diagnostic monitoring applies only to the current window and target (instance of a product).</td>
</tr>
</tbody>
</table>
Job status information

This chapter describes how to display job status information.

Displaying job status information

Tip

The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

The job status views provide access to status information and job output from the JES queues.

Use the following procedures to display:

- **JOBSTAT** - Displays status information about jobs, TSO users, and started tasks.
- **JOBLSTO** - Displays a list of output data sets that are associated with a job.

Note

As distributed, the job status views enable users to display information only about their own jobs. To grant users access to information about other jobs, TSO users, and started tasks, you must use the Job Status List resource definitions, as described in the MainView Security Reference Manual.

To display the JOBSTAT view

1. Perform one of the following actions:
   - Type **JOBSTAT** on the COMMAND line, and press Enter.
   - In Plex Manager, hyperlink from the Job Output List field on the EZPLEX menu.
By default, JOBSTAT displays information about any jobs that are owned by you (the TSO or MainView Explorer user). If you have permission, you can use the following filter fields to control the contents of JOBSTAT:

- **Owner ID**
- **Job Prefix**

Enter a specific value or a partial value followed by the * wildcard character, and press Enter. JOBSTAT is redisplayed with information about the specified owner IDs or jobs.

In the **CMD** field next to a job, you can enter one of the following line commands:

- **S** to display the output of the job.
- **P** to purge the output of the job.
- **DL** to display the output data sets for the job.

**To display the JOBLSTO view**

1. On the JOBSTAT view, perform one of the following actions:
   - Position the cursor on **JobName** and press Enter.
   - Type **DL** in the line command field next to a job and press Enter.

Figure 81 on page 270 shows an example of the JOBLSTO view.
JOBLSTO displays a list of output data sets that are associated with a job.

In the **CMD** field next to a ddname, you can enter the S line command to display the output of the data set.
Creating online Help

Customized Help text looks and works just like distributed Help text; both are displayed in scrollable pop-up windows.

Tip
The information described in this chapter applies to all MainView products except MainView FOCAL POINT.

You can create your own Help text for any view, any field within a view, or any topic. Depending on your needs, you can create all new Help topics or use the Help text that is distributed with your MainView product as a template.

Writing your own Help text

1. Create a partitioned data set (fixed block, LRECL=80) to serve as your private Help text library.

The preferred naming convention for this library is

userid.BBHDEF

where

- *userid* is your user ID.
- BBHDEF is the ddname that you will use to allocate this data set.

2. Allocate the data set to your TSO session by using the ddname, BBHDEF. (If you want each user to have access to the site-wide library, modify the MainView CLIST that you created during customization to allocate this data set for each user.)

3. Create a member in the data set with the same name as the view for which you want to write your own Help text.
To write your own Help text for a field, create a member in the BBHDEF data set with the same name as the view that contains that field.

For example, to create Help text for the JDELAY view or for a field contained within the JDELAY view, add a member called JDELAY to your Help text library.

4 Edit the member and add this tag to begin a Help topic:

:h1 id=xxxx.

where xxxx is the name of the view (if you are modifying view Help), the field's element name (if you are modifying field Help), or a topic ID value of 1 to 15 characters.

For example, to create Help text for the JDELAY view, add this tag to the member called JDELAY:

:h1 id=jdelay.

To create Help text for a field within the JDELAY view, add this tag to the JDELAY member:

:h1 id=xxxxx.

where xxxxx is the element name for that field.

To identify a field's element name, display the online Help for that field. The element name is displayed at the bottom of the pop-up window.

5 Add the rest of the tags to create your Help text by using the MainView tag language and control words.

For a description of the tag language and control words, see “Introduction to the MainView tag language” on page 275.

6 Save the member in your userid.BBHDEF library.

Note

The MainView tag language is a subset of the tags that are used by IBM BookMaster and BookManager. If you are familiar with these products, you may want to turn now to “MainView tags, attributes, and control words” on page 283 and start building your Help text immediately. If you need some background information, read “Introduction to the MainView tag language” on page 275 after you finish this topic.
How MainView locates your Help text

After you create your Help text, it is accessible to your TSO user ID during any MainView terminal session that is running in window mode.

When you issue the Help command for a view or field, the following process occurs:

1. MainView checks to see if you have a data set allocated with the name userid.BBHDEF. If you do, MainView looks for a member within that data set with the same name as the view for which you have requested Help. If the member is found, MainView searches the member's :h1 tags until it finds the id= attribute that matches the view or element name that you requested and displays the associated Help text in a Help pop-up window.

2. If MainView cannot find the appropriate member within your userid.BBHDEF data set, MainView checks to see if you have a site-wide library concatenated to your own user library and displays the associated Help text there, if it exists. (See the MainView Customization Reference for information about MainView product libraries and how site-wide libraries are created.)

3. If the Help text still has not been located, MainView displays the original Help text that is distributed in the load library hilevel.BBLINK.

The next topic introduces you to the MainView tag language and helps get you started on building your own Help text.

Introduction to the MainView tag language

The MainView tag language is a subset of the tags that are used by IBM BookMaster and BookManager.

This tag language controls how Help text is formatted in the MainView environment.

The topics in this section discusses some basic markup language concepts and describes the tags that you will need. However, this chapter is not intended to provide a comprehensive discussion on tag languages, nor a sophisticated set of examples. For more detailed information about a MainView tag, refer to the IBM BookMaster User’s Guide.

MainView also provides two control words, which are different from tags. Both the MainView tag language and the control words are discussed in this topic. For a complete list of the tags and control words supported by MainView, see “MainView tags, attributes, and control words” on page 283.
Using the MainView tag language

The following figure shows what the tags for a sample Help topic might look like.

**Figure 82: Example Help tags and text**

```html
/*This is a comment.
<h1 id=ASGDMN.Domain>
The Domain field identifies the SRM domain of the active address space.
</h1>
You can modify domain values by using the :hp2.SET DMN:ehp2. operator command.
<p>
The name of this element is ASGDMN.
```

This example illustrates the following issues:

- This topic is for a field with the element name ASGDMN, which is the id= attribute specified on the :h1 tag.
- The title of the pop-up window appears after the final period on the :h1 tag. In this case, the title is Domain.
- This topic is contained in the `userid.BBHDEF` member called ASGDMN.

When you place the cursor on the Domain field and press the Help PF key, the Help tags shown in Figure 82 on page 276 are formatted to look like Figure 83 on page 276.

**Figure 83: Formatted Help text**

<table>
<thead>
<tr>
<th>ddmmyyyy</th>
<th>hh:mm:ss</th>
<th>COMMAND</th>
<th>SCROLL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>W1</td>
<td>JDELAY=SYSB=*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Jobname</td>
<td>T</td>
<td>Dmn</td>
<td>Pg</td>
</tr>
<tr>
<td>RYS1SO</td>
<td>Help</td>
<td>Domain</td>
<td>4 17</td>
<td></td>
</tr>
<tr>
<td>MAK1</td>
<td>Command ==&gt;</td>
<td>Scroll ==&gt;</td>
<td>PAGE</td>
<td></td>
</tr>
<tr>
<td>CMR4X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSM1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSSMP1</td>
<td>The Domain field identifies the SRM domain of the active address space.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CATALO</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JES2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYSB</td>
<td>You can modify domain values by using the SET DMN operator command.</td>
<td>5 19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REC1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIR4</td>
<td>The name of this element is ASGDMN.</td>
<td>0 21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAOFRA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AAOGGS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CICSG3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPS2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LCM1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GIR2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBS1</td>
<td>T 20 223 2 73</td>
<td>2 44</td>
<td>0 29</td>
<td></td>
</tr>
<tr>
<td>GEN1</td>
<td>T 20 246 2 59</td>
<td>2 31</td>
<td>0 27</td>
<td></td>
</tr>
</tbody>
</table>

The name on the tag, Domain, appears as the title of this Help topic.
MainView tag language rules

When using the MainView tag language, you must adhere to the following rules:

■ Begin each tag with a colon (:) and end it with a period (.).

■ Use the paragraph tag (:p.) to place text that follows on a new line.

■ Add the title of the Help pop-up window after the period on the :h1 id= tag.

■ Enter tags in uppercase or lowercase letters, or as a combination of both. Tags are not case-sensitive.

■ Use end tags for the tags that require them. An end tag is exactly like its start tag partner, only the end tag adds the letter e immediately after the colon and before the tag name.

For example, in the Example Help tags and text figure in “Using the MainView tag language” on page 276, the words SET DMN are enclosed by a pair of highlighting tags. :hp2. turns bold text on and its end tag, :ehp2. , turns the bold off.

Note
All highlighting tags, :hp1., :hp3, :hp4, and so on, have the same result online as the :hp2. tag.

■ Use both optional and required attributes exactly as specified. An attribute is a word that appears after the colon and the tag itself, but before the final period.

For example, in the first line in the Example Help tags and text figure in “Using the MainView tag language” on page 276, :h1 is the tag, and id= is an attribute for that tag.

Attributes are used to modify some aspect of the tag, in the same way that parameters modify a command.

■ Separate attributes from their tags by a single blank space, but do not add space between the last attribute and the final period.

■ When two or more attributes are defined for a tag, include a single blank space before each attribute.

Using control words

Control words are macros that perform a specialized function.

Unlike tags, they do not control the format of text. Control words:
Must begin in the first column of an input line

- Begin with a . (period)
- Do not require a terminating character

The valid control words for MainView are shown in Table 29 on page 278.

Table 29: Valid control words

<table>
<thead>
<tr>
<th>Control word</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.*</td>
<td>Indicates that the text that follows is a comment. Comments do not appear in the Help text pop-up windows. In the example of a Help text member shown in the Example Help tags and text figure in “Using the MainView tag language” on page 276, the comment “This is a comment” is shown, but it does not appear in the formatted Help text shown in the Formatted Help text figure in “Using the MainView tag language” on page 276. If your comment exceeds one line, you must use the .* control word at the beginning of each line.</td>
</tr>
</tbody>
</table>
| .im           | Imbeds the text contained in the member name on the same line. Using the .im control word, you can imbed the same Help text in more than one Help member. For example, if you add .im ASGDMN to a new Help text member, the text shown in the Formatted Help text figure in “Using the MainView tag language” on page 276 will appear at the exact location where you placed the .im tag. The imbed tag is especially useful when you want to add your own note to a Help topic that was distributed with your MainView product. To add a note:

1. Create a member with an appropriate name.
2. Add your own text to the member.
3. Add an .im tag followed by the same id= value that you specified on your :h1 tag. The distributed MainView Help text is inserted wherever you placed the .im tag. (When you imbed a member into a member by the same name, MainView assumes that you mean to imbed the distributed Help text, not your own member. This process prevents recursive loops from occurring.) |

See “MainView tags, attributes, and control words” on page 283 for more information about the control words and their functions.

Commonly used tags

The most commonly used tags are:
The :h1 tag with its id= attribute
Use this tag to begin each new Help topic within a BBHDEF member.

The paragraph (:p.) tag, which leaves one line blank, and then begins a new line
You can also use the paragraph tag to insert a blank line between the :h1 tag and
the first paragraph that follows it.

The list tags, which are used to arrange discrete information units into one of four
different list formats

Using lists

Because you have a limited amount of room for text in a pop-up window, lists help
you present information economically.

MainView provides tags for four kinds of lists:

- Unordered lists
- Ordered lists
- Simple lists
- Definition lists

Unordered, ordered, and simple lists

An unordered list is simply a list of items with a bullet or dash next to each item, as
follows:

- Item A
- Item B
- Item C

An ordered list is the same as an unordered list, only numbers are used in place of
bullets. The numbering is handled for you automatically.

1. Item #1
2. Item #2
3. Item #3
Ordered lists are generally used to present a list of steps that a user must follow in the order presented.

A *simple list* is a list that has neither bullets nor numbers. Here is an example of a simple list:

Item A
Item B
Item C

The tagging for unordered, ordered, simple lists is similar. The following list describes the tags:

- :ul., :ol., or :sl. begins the list. The :ul. tag begins an unordered list, the :ol. tag begins an ordered list, and the :sl. tag begins a simple list. Place the tag on its own line.
- :li. marks a list item. Type the list item text on the same line immediately after the :li. tag.
- :eul., :esl., or :eol. ends the list. The :eul. tag ends an unordered list, the :esl. tag ends a simple list, and the :eol. tag ends an ordered list. Place the tag on its own line.

Figure 84 on page 280 shows the tagging for each type of list.

**Figure 84: Tagging for unordered, ordered, and simple lists**

```xml
<ul>
  <li>Jobname</li>
  <li>Jobtype</li>
</ul>

<ol>
  <li>Jobname</li>
  <li>Jobtype</li>
</ol>

<ol>
  <li>Jobname</li>
  <li>Jobtype</li>
</ol>
```

Figure 85 on page 280 shows the text that is produced by the tags shown in Figure 84 on page 280.

**Figure 85: Text produced by list tags**

```
ddmmmyyyy  hh:mm:ss ------- MainView WINDOW INTERFACE (Vv.r.mm) ---------
COMMAND ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>   
```
Definition Lists

Use definition lists to present a list of definitions in a two column format. The term, listed in the left column, is highlighted. The definition, listed in the right column, appears in regular text.

The tags for definition lists are different from those for ordered, simple and unordered lists. The tags are as follows:

- :dl. begins a list. Place this tag on its own line.
- :dt. denotes a term.
- :dd. marks the definition.
- :edl. ends the list. Place this tag on its own line.

**Figure 86: Tagging for a definition list**

```
:dl.
:dt.Jobname
:dd.Is a 1- to 8-character name for an address space.
:dt.Jobtype
:dd.Is the type of work an address space contains.
:edl.
```

The tags in Figure 86 on page 281 produce the definition list in Figure 87 on page 281.

**Figure 87: Output for definition list tags**

<table>
<thead>
<tr>
<th>Jobname</th>
<th>Is a 1- to 8-character name for an address space.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jobtype</td>
<td>Is the type of work an address space contains.</td>
</tr>
</tbody>
</table>

Commonly used tags
Creating free-form Help topics

In addition to view and field Help, MainView allows you to create free-form Help topics – topics that are not tied to a particular view or field.

Perhaps you want to document a condition that is specific to your site, for example, or add a topic that includes the names and telephone extensions of your site support personnel.

When creating your own Help text, proceed as follows:

- Supply any value that you choose on the id= attribute of the :h1 tag, up to 8 characters.
- Place the Help topic in a userid.BBHDEF member of the same name as defined on the id= attribute.

To display the Help topic, type HELP xxxx in any COMMAND field, where xxxx is the name of the member and the id= attribute value.

Setting up hypertext links between topics

You can establish hypertext links to move between Help text topics the same way that you use hyperlinks to move between views.

Hypertext links are created with the :link tag. When you use a link tag, the term that you associate with it is highlighted on your screen.

Before you establish a hypertext link, you need to know the id= value of the topic that you want to access.

The link tag has a matching end tag, and is coded as follows:

:link reftype=hd refid=xxxxx.term:elink.

where

- xxxx is the id= of the topic to which you want to hyperlink.
- term is the term that you want to highlight for that Help topic (such as its title).

Look again at the following Help topic:

:*This is a comment.
:h1 id=ASGDMN.Domain
:p.
The Domain field identifies the SRM domain of the active address space.

You can modify domain values by using the SET DMN operator command.

The name of this element is ASGDMN.

Suppose that you decide to establish a hypertext link between this topic and the topic that addresses the SET DMN command. The id= attribute for the SET DMN command is setdmn; therefore, you add the link tag to the Help topic as shown in Figure 88 on page 283.

**Figure 88: Sample link tag coding**

```
*.This is a comment
..h1 id=ASGDMN.Domin
..p.
The Domain field identifies the SRM domain of the active address space.
..p.
You can modify domain values by using the
..link reltype=hd refid=setdmn.SET DMN:elink.
operator command.
..p.
The name of this element is ASGDMN.
```

When you display this Help text on the screen, the link tag automatically displays the title, SET DMN, in bold or in reverse video (there is no need to enclose the term in :hp2. tags).

Hypertext links work just like hyperlinks. When you place the cursor on the highlighted term SET DMN and press Enter, Help on the SET DMN command is displayed.

---

## MainView tags, attributes, and control words

The MainView tags and control words are described in the first two tables of this topic. Each tag is described in the box labeled by the tag. For each tag, any end tags or attributes are also shown.

Any tag that is not supported is ignored by Help text. In the Help text pop-up window, the space occupied by an unsupported tag is compressed and does not appear.

Each table shows the syntax for the MainView tags, attributes, and control words. Each table contains:

- A tag with its related tags and attributes, if any
- An explanation of what the tags do and how to use them

Optional attributes are shown inside brackets ([ ]). Default values are underlined, and variables are italicized. For example, *nn* means any numeric value up to two...
digits. The vertical bar ( | ) indicates optional values for an attribute. You can use only one of the optional values for an attribute at a time.

Table 30: Definition list tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:dl [tsize=10</td>
<td>nn] [break=none</td>
</tr>
<tr>
<td></td>
<td>■ tsize=10</td>
</tr>
<tr>
<td></td>
<td>■ break=none</td>
</tr>
<tr>
<td></td>
<td>— none puts the description on the same line as the term even if the term is longer than the tsize value. none is the default.</td>
</tr>
<tr>
<td></td>
<td>— all puts the description on the line below its term in all cases.</td>
</tr>
<tr>
<td></td>
<td>— fit puts the description on the same line as the term unless the term is longer than the tsize value, in which case it puts the description on the line below.</td>
</tr>
<tr>
<td></td>
<td>■ compact formats the list without a blank line between the list items. If this attribute is omitted, a blank line is inserted between items.</td>
</tr>
<tr>
<td></td>
<td>■ thilite=y</td>
</tr>
<tr>
<td>:dthd.</td>
<td>(optional) Identifies the heading for the term column in a definition list (see :dl tag). It precedes the :ddhd. tag, with which it is paired. Both tags are valid only within a definition list.</td>
</tr>
<tr>
<td>:ddhd.</td>
<td>(optional) Identifies the heading for the description column in a definition list (see :dl tag). It is paired with the :dthd. tag, which should come first. Both tags are valid only within a definition list.</td>
</tr>
<tr>
<td>:dt.</td>
<td>Identifies the term that is being defined in a definition list (see :dl tag). This tag precedes the :dd tag with which it is paired. Both tags are valid only within a definition list.</td>
</tr>
<tr>
<td>:dd.</td>
<td>Identifies the description for the term that is being defined in a definition list (see :dl tag). It is paired with the :dt tag, which should come first. Both tags are valid only within a definition list.</td>
</tr>
<tr>
<td>:edl.</td>
<td>Required to end formatting of the definition list.</td>
</tr>
</tbody>
</table>
### Table 31: Caution and warning tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
</table>
| :caution [text=word]. | Alerts users to a risk of damage to equipment or data  
When preceded by a Paragraph tag (:p.), it places the highlighted word CAUTION after a blank line, followed by all the text (also highlighted) until the :ecaution. tag is encountered.  
text=word can be used to substitute another word or phrase for CAUTION. For example, :caution text='DANGER'. Do not move :ecaution. |
| :ecaution. | Required to end formatting of a caution message. |
| :warning [text=word]. | Alerts users to a possible error condition.  
When preceded by a Paragraph tag (:p.), it places the highlighted word Warning: after a blank line, followed by all the text (not highlighted) until the :ewarning. tag is encountered.  
text=word can be used to substitute another word or phrase for WARNING. For example, :warning text='DANGER'. Do not move :ewarning. |
| :ewarning. | Required to end formatting of a warning message. |

### Table 32: Figure, example, and lines tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:fig.</td>
<td>Turns formatting off. Under TSO/ISPF, it behaves identically to the :xmp. tag.</td>
</tr>
<tr>
<td>:efig.</td>
<td>Required to end formatting of a figure.</td>
</tr>
</tbody>
</table>
| :xmp.   | Turns formatting off.  
Text entered between the Example tag and an End Example tag (:exmp) is displayed exactly as it is entered, including leading blanks. Lines that are too long to fit in the window are clipped on the right margin. Example text is indented two spaces from the current left margin. |
| :exmp.  | Required to end formatting of an example.                                                                                                   |
| :lines. | Turns formatting off. Under TSO/ISPF, it behaves identically to the :xmp. tag.                                                             |
| :elines.| Required to end formatting after a Lines tag.                                                                                                |
Table 33: Heading level tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
</table>
| :h1 [id=topicid]. | Identifies the name of the element that contains the Help text. All text on the line following the :h1. tag until the next :h1. tag or :euserdoc. tag is formatted into a single Help pop-up window. The text on the same line as the :h1. tag is the title of the pop-up window, and it can be no longer than 38 characters. The title is fixed; the rest of the Help text is scrollable. Therefore, no other tags are allowed on the same line as the :h1. tag. If an :h1. tag is found within an imbed member (see the .im control word), it is treated as an :h2. tag.  

  id=topicid specifies the name of the view, element, or topic for which Help is being defined. The value defined to id= must be  
  - the name of the view (when defining view Help)  
  - the name of the element (when defining field or element Help)  
  - a unique value of 1 to 15 characters long for all other topics  

  The id is used to identify the online information when Help is requested and also can be used as the refid= value for hypertext links (see the :link. tag).  


Table 34: Highlight phrase tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
</table>

  For ISPF, the :hp2. highlight is high-intensity and is white on a color terminal. Everything else is low-intensity (blue) except for links (see the :link. tag). Therefore, :hp1. through :hp9. tags are allowed under ISPF but are treated as :hp2. tags.  

| :ehp1. through :ehp9. | Required to end formatting of a highlighted phrase. |
### Table 35: Link tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:link refid=element reftype=hd.</td>
<td>Establishes a link to hypertext. The word or phrase that follows the link tag appears turquoise in reverse video on a color terminal or high-intensity on a monochrome terminal. In all cases, the Tab key can be used to position the cursor to the term. refid=element specifies the name of another Help element to link to and display in another pop-up window. The ID that is specified in the refid= attribute must match the ID in the id= attribute of the :h1. tag for the element to which you want to link. reftype=hd must be entered as shown.</td>
</tr>
<tr>
<td>:elink.</td>
<td>Required to end a link reference.</td>
</tr>
</tbody>
</table>

### Table 36: Margin tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:lm [margin=nn].</td>
<td>Sets the left margin for the window. All text that follows is indented by the amount specified in the margin= attribute until another :lm. tag is encountered. If a left margin is not specified, the default margin is used. margin=nn determines the width of the indent. If omitted, the left margin is reset to the window width.</td>
</tr>
<tr>
<td>:rm [margin=nn].</td>
<td>Sets the right margin for the window. All text that follows is not formatted beyond the right margin column (which is calculated to be the window width minus the value specified in the margin= attribute) until another :rm. tag is encountered. margin=nn determines the width of the indent. If omitted, the right margin is reset to the window width.</td>
</tr>
</tbody>
</table>

### Table 37: General list tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:sl [compact].</td>
<td>Defines a nonsequential list of items. Each list item is indented without a preceding bullet or dash. compact suppresses the blank line between the items.</td>
</tr>
<tr>
<td>:ul [compact].</td>
<td>Defines a nonsequential list of items. Each list item is preceded by a bullet, or by a dash if it is nested within another unordered list. compact suppresses the blank line between the items.</td>
</tr>
<tr>
<td>:ol [compact].</td>
<td>Defines a sequential list of items and is usually used to list the steps in a procedure. Each list item is preceded by a sequential number, or by a letter if it is nested within another ordered list. compact suppresses the blank line between the items.</td>
</tr>
</tbody>
</table>
Any of the list types can be nested or imbedded within other lists. Each list is indented from the indented element that precedes it. When nesting lists, you must end each list with the appropriate list end tag when you finish that list. You must end formatting for the innermost list before ending formatting for a higher-level list.

If you use the compact attribute with a list tag and you want a blank line above the list, you must precede the list tag with a :p. tag. If you do not use the compact attribute and you precede a list tag with a :p. tag, two blank lines will precede the list. To remove one of the two blank lines, delete the :p. tag. One blank line will follow a list, whether or not the compact attribute is used and whether or not a :p. tag follows the End List tag.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:li.</td>
<td>Identifies the text that follows as an item within a list. The list item text is preceded by a sequential number if the item is part of an ordered list, by a bullet if it is part of an unordered list, or by nothing if it is part of a simple list. All text in a list item is blocked with each line indented just to the right of the number or bullet character. All other tags used within a list item are supported.</td>
</tr>
<tr>
<td>:lp.</td>
<td>Inserts a blank line and starts a new paragraph within a list item (:li.). It behaves exactly like the :p. tag. The formatted paragraph is flush with the list item text.</td>
</tr>
<tr>
<td>:esl:</td>
<td>Required to end a simple list.</td>
</tr>
<tr>
<td>:eol.</td>
<td>Required to end an ordered list.</td>
</tr>
<tr>
<td>:eul.</td>
<td>Required to end an unordered list.</td>
</tr>
</tbody>
</table>

### Table 38: Note tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:nt [text=word].</td>
<td>Inserts a note with indented text. The highlighted word Note: appears on the next line, followed by all text (not highlighted), including other tags, until the End Note tag (:ent.) is encountered. text=word substitutes another word or phrase for the word Note. For example follows, :nt text='BEWARE'.Do not exclude this field from the view:ent..</td>
</tr>
<tr>
<td>:ent.</td>
<td>Required to end formatting of a note.</td>
</tr>
<tr>
<td>:note [text=word].</td>
<td>Inserts a note as a single paragraph after a blank line. The highlighted word Note: appears on the next line, followed by all text (not highlighted) before the next tag is encountered. No end tag is required. text=word substitutes another word or phrase for the word Note. For example, :note text='BE CAREFUL'.If you exclude this field, the filter is still active.</td>
</tr>
</tbody>
</table>
Table 39: Paragraph tag

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:p.</td>
<td>Inserts a blank line in the output and starts a new paragraph. Two successive tags insert two blank lines, and so on.</td>
</tr>
</tbody>
</table>

Table 40: Parameter list tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:parml [tsize=10</td>
<td>nn] [break=none</td>
</tr>
<tr>
<td></td>
<td>tsizer specifies the space (where nn=some number of characters) to be allocated for the term column. The default is 10.</td>
</tr>
<tr>
<td></td>
<td>break=none</td>
</tr>
<tr>
<td></td>
<td>— none puts the description on the same line as the term even if the term is longer than the tsizer value.</td>
</tr>
<tr>
<td></td>
<td>— all puts the description on the line below its term in all cases. all is the default.</td>
</tr>
<tr>
<td></td>
<td>— fit puts the description on the same line as the term unless the term is longer than the tsizer value, in which case it puts the description on the line below.</td>
</tr>
<tr>
<td></td>
<td>compact suppresses the blank line between the list items. If this attribute is omitted, a blank line is inserted between items.</td>
</tr>
<tr>
<td>:pt.</td>
<td>Identifies the term being defined in a parameter list. It precedes the tag with which it is paired. Both tags are valid only within a parameter list.</td>
</tr>
<tr>
<td>:pd.</td>
<td>Identifies the description for the parameter being defined in a parameter list. This tag is paired with the tag, which should come first. Both tags are valid only within a parameter list.</td>
</tr>
<tr>
<td>:eparml.</td>
<td>Required to end formatting of a parameter list.</td>
</tr>
</tbody>
</table>

Table 41: Control words

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.* [comment text]</td>
<td>Allows you to place comments in the input file. Comments are not formatted into the output window and are used merely to place informative notes within your source files. The comment control word (.* ) must begin each line that has comments.</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.im <code>filename</code></td>
<td>Allows you to include source input text that resides in another member of the same PDS or in the <code>hilevel.BBSAMP</code> data set, which contains MainView Help text in its original form. The name of the member (or the element name) follows the imbed control word, separated by at least one blank. If the requested member is not found, a warning message is inserted in the formatted output. The imbed statement occupies the entire input line. The only limit to the nesting level of imbeds is the total size of the output buffer.</td>
</tr>
</tbody>
</table>

For additional information about control words, see “Using control words” on page 277.
BBI diagnostic facilities

**Tip**
The information described in this appendix applies to the following products only:

- MainView FOCAL POINT
- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for MQ

Because of the many environments in which BBI code executes, the BBI diagnostic facilities perform differently depending on the type of error and where it is encountered.

When an abend occurs in BBI or in a module of a product that uses a BBI module, recovery procedures gain control, collect the required information for problem analysis, write messages and diagnostics, and attempt recovery. Only if recovery fails or a specified recovery limit count is reached will the address space terminate and activate standard system or subsystem dumping facilities (SYSUDUMP, SYSABEND, SYSMDUMP).

The level of diagnostic information produced by default is that necessary to solve the majority of problems. Additional dump, trace, and display facilities are available if this information is not sufficient. In such a case, you might be asked by Customer Support personnel to collect more information.

Recovery procedures often limit the effect of errors occurring in BBI code. Because the temporary loss of data or function can be unnoticed or unimportant at the time, these errors are not always reported to the BBI system administrator. To prevent future errors, perhaps at a more critical point, operational procedures should be set
up to check for such occurrences. Because a software log record (LOGREC) is usually produced for each BBI abend, a regularly scheduled job to print off BBI LOGRECs is the easiest method to use; see “LOGREC” on page 292.

Messages

If possible, messages are written documenting the error and the actions taken.

Depending on where the error occurs and which functions are accessible, the messages can be written to one or more of these destinations:

- SYSLOG (WTO) (has the BBI-SS PAS subsystem ID appended)
- BBI-SS PAS journal log
- terminal session journal log
- short message in the upper right corner of the screen

Typical messages for an abend condition are

<table>
<thead>
<tr>
<th>xxnnnnc</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBI/xx</td>
<td>STAE EXIT ENTERED</td>
</tr>
<tr>
<td>BBI</td>
<td>SOFTWARE RECORD LOGGED TO LOGREC</td>
</tr>
<tr>
<td>DUMP</td>
<td>IS NOW BEING TAKEN</td>
</tr>
<tr>
<td>RETRY</td>
<td>WILL BE ATTEMPTED</td>
</tr>
</tbody>
</table>

where xx varies by component.

All messages, including short messages, are documented in the MESSAGES application; use the MESSAGES option from the Primary Option Menu or browse the BBMLIB data set.

LOGREC

A software record (LOGREC) is logged to SYS1.LOGREC when a BBI abend occurs, if conditions permit.

Although this method is a shorthand method of recording abends, it contains the location of the error, abend code, and registers. Often this information is enough to solve the problem. All LOGRECs written by BBI are identified with BBI so they can be selectively viewed.

To print BBI LOGRECs, run the job in BBSAMP member LOGRECS. The JCL stream is set up to print only LOGRECs produced by BBI. This job only reads; it does not
reset the SYS1.LOGREC data set, so it does not interfere with any data center procedures. However, if you regularly clean out this data set, schedule your job to run before the clean up occurs (preferably in the same job stream).

If you own MainView SYSPROG Services, you can also browse these LOGRECs online. The command format is

**SOF,BBI,MAP**

You can also use MainView SYSPROG Services to check for the presence of BBI LOGRECs before running the batch job to print them.

## Dumps

In most abend conditions, BBI writes a dump in addition to the LOGREC.

Parameters in various BBPARM members control whether dumping is the default for the different environments (DUMPS=YES|NO). The type of dump also varies according to where the abend occurs.

### Types produced

The types of dumps that can be produced are as follows:

- **SNAP dump**
  
  This dump is usually produced as the first part of a BBI-formatted dump. The SNAP is issued only in the BBI address spaces (BBI-SS PAS and UAS), not in the target system.

  It is in the standard OS dump format, showing the PSW at abend, ASCB, TCB, CDEs, registers, RTM2WA, system trace table, and so forth. The most useful information is in the RTM2 Work Area (completion code, abending program name, PSW, registers at abend). It does not include any private storage.

  Refer to the following section on the BBI-formatted dump for a description of where this dump is written.

- **BBI-formatted dump**
  
  This dump is a short-form dump formatted to make the most important information easy to find and read. It is produced in BBI address spaces (BBI-SS PAS and UAS) as well as under the subtasks in the target regions.

  The formatted dump displays the System Diagnostic Work Area (SDWA), failing program information (PSW, ILC, INTC, TEA, name of program), registers at
abend (and at the next RB level if different), the storage around the PSW and register locations, and selected BBI control blocks.

This type of dump is written to a special DDNAME BBIDUMP rather than to the standard SYSUDUMP. The BBI dump data set is not defined in the JCL, but is allocated dynamically at startup with the SYSOUT class specified in BBISSP00 with the parameter DUMPCLAS. The default value is W.

When the data set is allocated, the BBI banner page identifying the dump is written to it, ready for the first dump. (This process avoids the loss of current data at abend while doing dump formatting.)

Once completed, the dump is spun off for immediate access and the dump data set is reallocated and primed.

- **SVC dump**

  An SVC dump is taken when the DUMPS option specifies SDUMP or ALL. DUMPS=SDUMP is the recommended setting to obtain all diagnostic information at the time of the first occurrence of a problem. A LOGREC record is always written and it is usually sufficient to identify the problem and sometimes resolve it.

  The SVC dump request will dump all of CSA, private storage for the BBI-SS PAS, and any secondary address spaces if the abend occurred in cross-memory mode.

  An SVC dump is written to a SYS1.DUMPxx data set, and a message is sent to the console operator. Copy the dump data set to another data set and clear the original SYS1.DUMPxx data set. Dump symptoms are set to prevent multiple dumps of the same problem being taken.

  Contact BMC Customer Support for help in further analyzing the problem. They will request the dump, if required, or the LOGREC if it is sufficient to resolve the problem.

- **SYSUDUMP / SYSABEND**

  Standard dumps to the regular SYSUDUMP or SYSABEND DDs are written only if BBI recovery fails (or the specified recovery limit count was reached with BACKOUT=NO), which causes the affected address space to be terminated.

**BBI dump parameters**

BBI dumping procedures are controlled by various parameters specified in BBPARM for the different running environments.

Refer to these members for additional information about the DUMPS parameter.

- **BBISSP00**

  This member has a DUMPS parameter, with the default setting of SDUMP. This controls specific BBI routines in the BBI-SS PAS and in the product subtasks.
DUMPS=YES specifies that a BBI-formatted dump should be taken. In the BBI-SS PAS, the formatted dump is preceded by a SNAP dump. BBISSP00 has parameters to specify the SYSOUT class for the BBI dumps (DUMPCLAS).

- BBITSP00 in BBPROF
  The DUMPS parameter controls BBI-formatted dumping in that user address space. The default is YES. The formatted dump is preceded by a SNAP dump.

- BBIXSP00
  The DUMPS parameter in this BBPARM member controls SVC dumps under BBI SRB routines. The default is NO to avoid unnecessary overhead.

**Summary of BBI abend diagnostics**

Abends in BBI address spaces produce the following diagnostics:

- WTO messages
- messages to the BBI-SS PAS or terminal session journal log
- LOGREC
- BBI-formatted dump including SNAP information about DD BBIDUMP if BBISSP00 parameter DUMPS=YES (default)

Abends in routines that run in SRB mode to collect data from a target produce the following diagnostics:

- short message to terminal session user: SRB ABENDED
- messages on the BBI-SS PAS journal log that a service failed
- LOGREC
- SVC summary dump, if a special parameter is activated for additional diagnostics
Journal and image log record formats

This appendix describes the journal log and image log record formats.

Tip
The information in this appendix applies to specific MainView products as described in each section.

Journal log record formats

The following table defines the print record format for the journal log as produced by the sample procedures DLOGJCL or PLOGJCL.

Tip
The information in this section applies to the following products only:

- MainView FOCAL POINT
- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online
- MainView for MQ

The DCB attributes are

DSORG=PS,LRECL=121,RECFM=FBA
Table 42: Journal log print record format

<table>
<thead>
<tr>
<th>Column</th>
<th>Length</th>
<th>Record description</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>1</td>
<td>blank</td>
</tr>
<tr>
<td>002</td>
<td>6</td>
<td>Julian date, in the format yy.ddd</td>
</tr>
<tr>
<td>008</td>
<td>3</td>
<td>blanks</td>
</tr>
<tr>
<td>011</td>
<td>8</td>
<td>time stamp, in the format hh:mm:ss</td>
</tr>
<tr>
<td>019</td>
<td>1</td>
<td>blank</td>
</tr>
<tr>
<td>020</td>
<td>8</td>
<td>system ID (SSID) or user identifier (TSID, jobname, and so on)</td>
</tr>
<tr>
<td>028</td>
<td>1</td>
<td>blank</td>
</tr>
<tr>
<td>029</td>
<td>76</td>
<td>message or command text</td>
</tr>
<tr>
<td>105</td>
<td>17</td>
<td>blanks</td>
</tr>
</tbody>
</table>

Image log record formats

The BBI-SS PAS and terminal session image logs in BBI record screen image format are shown in the following table.

Tip

The information in this section applies to the following products only:

- MainView FOCAL POINT
- MainView AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS Online

The records contain a complete screen layout and are 2360 bytes in length; their format is described in this table and in BBSAMP member IMFMSG.

Table 43: Screen image record format

<table>
<thead>
<tr>
<th>Offset</th>
<th>Name</th>
<th>Length</th>
<th>Record description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RTOMSG</td>
<td>DSECT</td>
<td></td>
</tr>
<tr>
<td>Offset</td>
<td>Name</td>
<td>Length</td>
<td>Record description</td>
</tr>
<tr>
<td>--------</td>
<td>----------------</td>
<td>----------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>000000</td>
<td>MSGLLZZ</td>
<td>DS XL4</td>
<td>message LLZZ field / log RDW</td>
</tr>
<tr>
<td>000004</td>
<td>MSGLOGCD</td>
<td>DS XL1</td>
<td>log record code = X'FB'</td>
</tr>
<tr>
<td></td>
<td>MSGCON</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'FB'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000005</td>
<td>MSGTYPE</td>
<td>DS XL1</td>
<td>type of record</td>
</tr>
<tr>
<td></td>
<td>MSGDSPLY</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'40'</td>
<td></td>
<td>display (information/function)</td>
</tr>
<tr>
<td></td>
<td>MSGWARN</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'80'</td>
<td></td>
<td>warning message</td>
</tr>
<tr>
<td>000006</td>
<td>MSGMODE</td>
<td>DS XL1</td>
<td>mode of creation</td>
</tr>
<tr>
<td></td>
<td>MSGSYNC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'40'</td>
<td></td>
<td>sync request from terminal</td>
</tr>
<tr>
<td></td>
<td>MSGASYNC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'80'</td>
<td></td>
<td>async request from BBI-SS PAS</td>
</tr>
<tr>
<td></td>
<td>MSGDM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'20'</td>
<td></td>
<td>RTO message created by Data Manager</td>
</tr>
<tr>
<td></td>
<td>MSGNOLOG</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'01'</td>
<td></td>
<td>do not log async request</td>
</tr>
<tr>
<td>000007</td>
<td>MSGRTOCD</td>
<td>DS XL1</td>
<td>realtime service completion code</td>
</tr>
<tr>
<td></td>
<td>MSGCCOK</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'00'</td>
<td></td>
<td>successful</td>
</tr>
<tr>
<td></td>
<td>MSGCCNV</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQU X'01'</td>
<td></td>
<td>invalid request</td>
</tr>
<tr>
<td>000008</td>
<td>MSGSELRNR</td>
<td>DS H</td>
<td>select number (used only in IMRPRINT) (initialized to zero)</td>
</tr>
<tr>
<td>00000A</td>
<td>MSGDATE</td>
<td>DS PL4</td>
<td>date - packed decimal '00yydddf'</td>
</tr>
<tr>
<td>00000E</td>
<td>MSGTIME</td>
<td>DS PL4</td>
<td>time - packed decimal 'hhmmssstf'</td>
</tr>
<tr>
<td>000012</td>
<td>MSGLTERM</td>
<td>DS CL8</td>
<td>LTERM name of service requestor</td>
</tr>
<tr>
<td>00001A</td>
<td>MSGPLINE</td>
<td>DS XL2</td>
<td>physical line number</td>
</tr>
<tr>
<td>00001C</td>
<td>MSGPTERM</td>
<td>DS XL2</td>
<td>physical terminal number</td>
</tr>
<tr>
<td>00001E</td>
<td>MSGLINE2</td>
<td>DS 0CL33</td>
<td>second line of request/response</td>
</tr>
<tr>
<td>00001E</td>
<td>MSGFNC</td>
<td>DS 0CL7</td>
<td>function code MFLD</td>
</tr>
<tr>
<td>00001E</td>
<td>MSGFNCAT</td>
<td>DS CL2</td>
<td>dynamic attribute bytes</td>
</tr>
<tr>
<td>000020</td>
<td>MSGFNCDDT</td>
<td>DS CL5</td>
<td>service select code</td>
</tr>
<tr>
<td>000025</td>
<td>MSGTLE</td>
<td>DS 0CL26</td>
<td>screen title MFLD</td>
</tr>
<tr>
<td>000025</td>
<td>MSGTLEAT</td>
<td>DS CL2</td>
<td>dynamic attribute bytes</td>
</tr>
<tr>
<td>000027</td>
<td>MSGTLEDT</td>
<td>DS CL24</td>
<td>service title</td>
</tr>
<tr>
<td>00003F</td>
<td>MSGLINE3</td>
<td>DS 0CL81</td>
<td>third line of request/response</td>
</tr>
<tr>
<td>00003F</td>
<td>MSGSYS</td>
<td>DS 0CL81</td>
<td>system message field</td>
</tr>
<tr>
<td>00003F</td>
<td>MSGSYSAT</td>
<td>DS CL2</td>
<td>dynamic attribute bytes</td>
</tr>
<tr>
<td>Offset</td>
<td>Name</td>
<td>Length</td>
<td>Record description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------</td>
<td>--------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td>000041</td>
<td>MSGSYSDT</td>
<td>DS CL55</td>
<td>input parm/msg/request-ID</td>
</tr>
<tr>
<td></td>
<td>MSGSYSM</td>
<td>EQU</td>
<td>message area</td>
</tr>
<tr>
<td></td>
<td>MSGSYSDT</td>
<td>+6,49,C'C'</td>
<td></td>
</tr>
<tr>
<td>000078</td>
<td></td>
<td>DS 0CL24</td>
<td>reserved for Performance Management panel</td>
</tr>
<tr>
<td>000088</td>
<td>MSGIMSN</td>
<td>DS CL8</td>
<td>IMS name for IMRPRINT</td>
</tr>
<tr>
<td>000090</td>
<td>MSGPFKIN</td>
<td>DS 0CL84</td>
<td>PF key input area</td>
</tr>
<tr>
<td></td>
<td>MSGPFKSC</td>
<td>DS CL5</td>
<td>service select code</td>
</tr>
<tr>
<td></td>
<td>MSGPFKSP</td>
<td>DS CL79</td>
<td>service parm</td>
</tr>
<tr>
<td></td>
<td>ORG MSGPFKIN</td>
<td></td>
<td>reset origin</td>
</tr>
<tr>
<td>000090</td>
<td>MSGWMSG</td>
<td>EQU</td>
<td>warning message</td>
</tr>
<tr>
<td></td>
<td>MSGWLEN</td>
<td>EQU</td>
<td>length of warning message log record</td>
</tr>
<tr>
<td></td>
<td>MSGWLEN</td>
<td>*-RTOMSG+84</td>
<td></td>
</tr>
<tr>
<td>000090</td>
<td>MSGDATA</td>
<td>DS 0CL2112</td>
<td>display area cleared to blanks</td>
</tr>
<tr>
<td>000090</td>
<td>MSGLINES</td>
<td>DS 21CL81</td>
<td>21 lines - 2 attr + 79 data</td>
</tr>
<tr>
<td>000735</td>
<td>MSGFREE</td>
<td>DS CL483</td>
<td>work area for each service</td>
</tr>
<tr>
<td></td>
<td>MSGLEN</td>
<td>EQU</td>
<td>length of RTO message is X '918'</td>
</tr>
<tr>
<td></td>
<td>MSGLEN</td>
<td>*-RTOMSG</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MSGLNNUM</td>
<td>EQU</td>
<td>specify 21 lines in display area</td>
</tr>
<tr>
<td></td>
<td>MSGLNLEN</td>
<td>EQU</td>
<td>81 byte lines - 2 attr + 79 data</td>
</tr>
<tr>
<td></td>
<td>MSGLNLEN</td>
<td>81</td>
<td></td>
</tr>
</tbody>
</table>
BMC Subsystem Services

The BMC Subsystem Services (BBXS) are a set of common service routines that are used by the following BMC products:

- CMF MONITOR
- MainView AutoOPERATOR for z/OS
- MainView for IMS Online
- MainView for z/OS
- MainView VistaPoint

This appendix provides an overview of BBXS and describes installing and initializing the subsystem in a system image or in a system running under VM. Understanding BBXS functions and initialization procedures is necessary because the subsystem resides in CSA (common service area) memory and can impact your system performance if not installed, initialized, and used properly.

Overview of BBXS

BBXS has three parts:

- formal subsystem
- service routines that are dynamically loaded into extended CSA
- extended CSA memory used for
  — control table (BBCT)
  — subsystem service routines
  — shared data areas
Some of this storage is page-fixed.

From an external view, BBXS is a proper subsystem. However, unlike subsystems such as JES2 and JES3, BBXS does not require its own address space. BBXS uses only the **SSCTSUSE** field of the subsystem's CVT or SSCVT to anchor the BMC Subsystem Services Control Table (BBCT). The BBCT is built during BBXS initialization, and primarily contains pointers to BBXS service routines. These routines return data in response to specific requests from BMC products.

**Using BBXS with multiple BMC products**

Because many BMC products use BBXS, it is important that they all use the same version of BBXS. This process avoids errors that can occur when a down-level version of the CSA-resident components of BBXS attempts to run with a more recent version of the dynamic BBXS components or vice versa. When maintenance is applied to BBXS or a product that requires BBXS, it can affect all products that use BBXS.

**Installing BBXS**

Installing the subsystem is part of the installation process for any product that requires BBXS.

The BBXS FMID, BBBBXnn, where *nn* is the current BBXS version and release level, is included in the FMID set of every BMC product that requires BBXS.

Place all BMC product load modules, including the BBXS load modules, in the *prefix.BBLINK* data set. The BBLINK data set must be APF-authorized. If you use BBLINK from either STEPLIB or JOBLIB, all BMC products must specify the same data set.

BMC recommends that you place the BBLINK data set in the LINKLST library concatenation. Using LINKLST is important, but it is even more important to use only one initialization method and to control it carefully.
Initializing BBXS

BMC ships a sample started task procedure for BBXS initialization in the `prefix.BBSAMP` data set.

During customization, the `@BBXINIT` member is copied to a library that you specify and renamed to `BBXSINIT`. This procedure runs the `BBXSINIT` program that initializes or reinitializes the BBXS subsystem.

BMC recommends that you use the `BBXSINIT` procedure early in the IPL process or after applying maintenance to BBXS.

---

**Note**

Except in places where the `@BBXINIT` procedure is specifically mentioned, the remainder of this appendix uses the term `BBXSINIT` interchangeably to refer to both the program and the procedure that executes the program. After BBXS is initialized, routines and shared data areas loaded into CSA memory remain there until the next IPL.

The `BBXSINIT` program, as the name implies, creates new copies of the BBXS modules and data areas in CSA and initializes all ongoing BBXS processes, such as channel and device data collection. Any new product startups or product restarts use the new BBXS code and data areas. CSA used by a previous initialization of BBXS is not released until an IPL occurs.

Do not run the `BBXSINIT` program as part of another startup procedure. If the other procedure needs to run more than once between IPLs, BBXS is reinitialized and claims more valuable CSA space.

Reinitialization of BBXS is required so that changes to CSA-resident BBXS modules become effective and dynamically loaded modules match the version of the CSA-resident modules. The reinitialized version of BBXS works with all versions of BMC products because BBXS is downwardly compatible.

---

**Note**

- BMC recommends that you recycle all products that use BBXS after the BBXS subsystem is reinitialized so that the new BBXS code is used by all applications.

- The COMMON STORAGE MONITOR component of MainView for z/OS allows the user to collect data on the allocations and use of common storage (CSA). This component can be activated or not activated. BMC recommends starting this monitor as soon as possible in order to track all CSA allocations directly after IPL.
Working with DB2 Component Services (DBC)

This chapter explains how to start and manage the DBC subsystem and how to manage DBC security.

Working with the DBC subsystem

To use the DBC technology, you must start and manage a DBC subsystem.

This section explains how to:

- Start a DBC subsystem
- Specify DBCPARMS parameters
- Stop a DBC subsystem

Note
If you are using a version of IBM z/OS earlier than Version 1 Release 12, ensure that the AF_UNIX MAXSOCKETS value in BPXPRMxx is set to a value of at least 1000 before starting the DBC subsystem. For more information about the MAXSOCKETS value, refer to IBM documentation.

Started task for the DBC subsystem

Normally, you start the DBC subsystem as a z/OS started task.
When performing an IPL, wait until after OMVS initialization has completed to start the DBC. Otherwise, DBC might experience failures as OMVS initialization executes and might produce the following messages:

BMC24549 XA00 Error on DBC INIT, LOGSET=XA00L005,NGL AGENT=XA00, RC=0000000C
BMC24948 XA00 LOGSET XA00L005 terminating

During OMVS initialization, wait to start the DBC subsystem until the following message is displayed:

D1 N 4000000 AO 12232 02:33:03.95 00000010 BPX10041 OMVS
INITIALIZATION COMPLETE

You should add the JCL procedure for the started task into a system procedure library.

**Note**

For testing or trial installations, you can also start the DBC subsystem as a batch job. However, the JES initiator will be busy for the life of the DBC subsystem. BMC does not recommend this approach for non-trial installations.

The following figure shows an example of the started task for DBC.

**Figure 89: JCL procedure for the DBC started task**

```plaintext
/*******************************************************************************/
/* Description: */
/* BMC Software DBC subsystem JCL procedure for the started task. */
/* */
/* Customization Steps: */
/* - Modify the DBC subsystem initialization parameters in the */
/* member DBC$PARM that the DBCPARMS DD statement identifies. */
/* - Modify the DBC subsystem security parameters in the member */
/* DBC$SECU that the DBCSECUR DD statement identifies. */
/* - Allocate Registry data set for products running under DBC */
/* Sample define: */
/*   DEFINE CLUSTER (NAME(BMCDBC.ssid.REGISTRY) - */
/*     LINEAR CYL(25 10) SHAREOPTIONS(1,3) STORCLAS(xxxxxx)) */
/* - Add this JCL procedure to a system procedure library. */
/* - APF authorize the DBC STEPLIB data set. */
/* */
/* Notes: */
/* The DBC subsystem is a long-running-service address space that */
/* normally remains active for the life of an IPL. Therefore, BMC */
/* does not recommend starting the DBC subsystem as a batch job. */
/* Doing so causes the JES initiator to be busy for the life of */
/* the DBC subsystem. If you want to run the DBC as a batch */
/* job, replace the PROC statement with a valid JCL job card. */
/* */
/*******************************************************************************/
/DBC      PROC SSID=,GRP=,REPMODE=
/DBC      EXEC PGM=DBCMAIN,REGION=0M,ACCT=&ACC,TIME=&TIM,
/STEPLIB  DD DSN=BMCPERF.TEST.BMCLINK,
/DBCP aims DD SYSOUT=*,RECFM=VA
/SYSPRINT DD SYSOUT=*,RECFM=VA
```
//BMCPSWD DD DISP=SHR, DSN=BMCPERF.TEST.BMCLINK  
//SYSTSPRT DD SYSOUT=*  
//SYSTERM DD SYSOUT=*,RECFM=VA  
//DBCPARMS DD DISP=SHR, DSN=BMCPERF.TEST.BMCLINK  
//DBCSIRT DD DISP=SHR, DSN=BMCPERF.TEST.BMCLINK  
//DOMTMPLT DD DISP=SHR, DSN=BMCPERF.TEST.BMCTMPLT  
//DOMPARMS DD DISP=SHR, DSN=BMCPERF.TEST.BMCPARM

Table 44 on page 307 describes DD statements that you define within the started task.

**Table 44: DD statements for the DBC started task**

<table>
<thead>
<tr>
<th>DD statement</th>
<th>Identifies the</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>DBC load library</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The DBC STEPLIB load library must be APF authorized.</td>
</tr>
<tr>
<td>DBCPRINT</td>
<td>Data set to dynamically write DBC messages</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>Data set to write potential non-DBC product application messages</td>
</tr>
<tr>
<td>BMCPARMS</td>
<td>Data set containing the product license keys</td>
</tr>
<tr>
<td>DBCPARMS</td>
<td>DBC initialization parameters file</td>
</tr>
<tr>
<td></td>
<td>For more information, see “DBC startup parameters” on page 307.</td>
</tr>
<tr>
<td>DBCSECUR</td>
<td>Security parameters file</td>
</tr>
<tr>
<td></td>
<td>For more information, see “DBC security parameters” on page 325.</td>
</tr>
<tr>
<td>DOMTMPLT</td>
<td><em>(for System and SQL Performance products only)</em> Data set that contains advisor text</td>
</tr>
<tr>
<td>DOMPARMS</td>
<td><em>(for System and SQL Performance products only)</em> Data set that contains parameters for advisors</td>
</tr>
</tbody>
</table>

**DBC startup parameters**

The DBC subsystem has the following sets of logically discrete startup parameters:

- **DBCPARMS** parameters include the required DBC subsystem ID (SSID) and optional parameters, such as the XCF group name and repository data set name. You specify this general set of control parameters through the DBCPARMS DD statement. For more information, see “DBCPARMS parameters” on page 308.

- **DBCSECUR** parameters relate specifically to security customization. You identify these parameters through the DBCSECUR DD statement. DBC maintains the security parameters as a discrete set of parameters so that you can implement a
different level of data set security for these parameters, if needed. For more information, see “DBC security” on page 320.

**Note**
The only required DBC initialization parameter is the SSID; along with the XCF group name, you can also specify the SSID via the EXEC PGM=DBCMAIN,PARM=’params’ JCL statement. If you specify the DBC SSID or GROUP through the JCL PARM= override statement, these values take precedence over any <SSID> or <GROUP> XML element values specified in DBCPARMS statement.

Figure 90 on page 308 shows an example of the started task JCL with parameters.

**Figure 90: Started task for DBC**

```
//DBC      PROC
//DBCEXEC  EXEC PGM=DBCMAIN,PARM='SSID=DO&SYSCLONE,GROUP=DBCGROUP,REPMODE='
//DBCxxx10 EXEC PGM=DBCMAIN,REGION=0M,ACCT=xxxx,TIME=1440,
//PARM=('SSID=&SSID,MODE=&REPMODE,GROUP=&GRP')
//STEPLIB  DD DISP=SHR,DSN=BMC.DBC.BMCLINK
//SYSPRINT DD SYSOUT=*,RECFM=VA
//DBCPARMS DD DISP=SHR,DSN=BMC.DBC.UBMCSAMP(DBC$PARM)
//DBCSECUR DD DISP=SHR,DSN=BMC.DBC.UBMCSAMP(DBC$SECU)
```

**DBCPARMS parameters**
The DBCPARMS DD JCL statement identifies the initialization parameters file for the DBC started task.

**Note**
Initialization parameters are separated from the DBC security startup parameters that you specify in the DBCSECUR DD statement. This separation allows you to manage the security parameters separately from other DBC parameters by implementing data set name security through your External Security Manager (ESM).

For more information about the started task, see “Started task for the DBC subsystem” on page 305. For more information about security parameters, see “DBC security parameters” on page 325.

**Guidelines for sharing DBCPARMS across multiple DBC subsystems**
If you want to share a single DBCPARMS parameters file across all DBC subsystems, consider the following guidelines:
The DBC SSID must be unique within an XCF group and unique on a single LPAR. You must specify the required DBC SSID by using the JCL parameter override statement (`EXEC PGM=DBCMAIN,PARM='SSID=ssid')

You can use the `&SYSCLONE` symbolic from the IBM MVS system symbols as part of the SSID= parameter. Doing so allows you to create unique DBC SSIDs across the sysplex without having separate PROCs. You can create two-, three-, or four-character IDs by combining `&SYSCLONE` with other literal characters (for example, `SSID=DC&SYSCLONE`). To determine the value of `&SYSCLONE` at your site, contact your system programmer.

By default, the DBC starts an XCF group based on the name of the DBC group. You can specify the DBC group name by using the JCL parameter override statement (`EXEC PGM=DBCMAIN,PARM='GROUP=group')`. You can specify the XCF group name by using the `<XCFGROUP>` element in the DBCPARMS member.

You can specify the DBC repository data set name in the DBCPARMS parameters file by using one or more DBC system variables. Doing so ensures a unique repository data set name for each DBC subsystem instance. Table 45 on page 309 lists the DBC system variables that you can use to create a unique data set name.

**Table 45: DBC system variables**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;SSID.</td>
<td>4-byte subsystem ID of the DBC subsystem</td>
</tr>
<tr>
<td>&amp;SMFID.</td>
<td>4-byte SMF ID of the host system</td>
</tr>
<tr>
<td>&amp;MVSNAME.</td>
<td>8-byte system name of the host system</td>
</tr>
</tbody>
</table>

For more information about the individual DBC initialization parameters, see “DBCPARMS elements” on page 310.

**Structure of the XML stream**

Figure 90 on page 308 shows a sample parameters file.

```
<DBC>.......
<SSID>SSID</SSID>
<GROUP>BMCGROUP</GROUP>
<XCFGROUP>DBCXCFF</XCFGROUP>
<MODE>NORMAL</MODE>
<WToupperCase>YES</WToupperCase>
```
For JCL sample files, see the SAMP data set.

**DBCPARMS elements**

*(required)* The `<DBCPARMS>` element is the root-level element in the XML stream that is used to define DBC parameters.

**Data type:** Not applicable.

**Child elements:** `<OPTIONS>`

**OPTIONS**

*(required)* The `<OPTIONS>` element contains the DBCPARMS values.

**Data type:** Not applicable.

**Parent element:** `<DBCPARMS>`

**Child elements:** `<SSID>`, `<GROUP>`, `<WTOUPPERCASE>`, `<XCFGROUP>`, `<MODE>`, `<WLM>`, and `<DPRREPOS>`

**SSID**

*(optional)* The `<SSID>` element specifies the SSID of the DBC subsystem. The SSID is a required DBC startup parameter. You must specify the SSID either through the DBCPARMS XML document or in the EXEC statement; otherwise, initialization fails.

Consider the following requirements for the value of the `<SSID>` element:

- BMC recommends that you use different names for the DBC SSID and the DBC started task.
- The SSID value must not conflict with any MVS command verbs. If it does, DBC issues an error message and terminates.
The SSID value can contain A–Z, 0–9, #, or $. Any other characters are invalid and cause the DBC subsystem to issue an error message and terminate.

The SSID value should not begin with a numeric character (0–9). You can use the SSID to issue system commands to the DBC subsystem; consequently, if the SSID value started with a number, the system could misinterpret the value as the short form of the REPLY system command. (For more information about issuing system commands, see “DBC commands” on page 317.)

The SSID value should not conflict with any JES commands. This recommendation ensures that only the DBC subsystem processes system commands that you issue to the DBC subsystem by using the DBC SSID value.

Data type: VARCHAR(4)
Parent element: <OPTIONS>
Child elements: None.

GROUP

(optional) The <GROUP> element specifies the name of the DBC XCF group that relates DBC subsystems on different LPARs within the SYSPLEX.

Note
You can also specify the DBC XCF group through the EXEC PGM=DBCMAIN,PARM='parms' JCL statement.

Consider the following requirements for the value of the <GROUP> element:

- The GROUP value must not conflict with any MVS command verbs. If it does, DBC issues an error message and terminates.
- The GROUP value must not contain embedded blank characters.
- The GROUP value should not begin with a numeric character (0–9). You can use the group name to issue system commands to the DBC subsystem; consequently, if the group name started with a number, the system could misinterpret the value as the short form of the REPLY system command. (For more information about issuing system commands, see “DBC commands” on page 317.)
- The GROUP value should not conflict with any JES commands. This recommendation ensures that only the DBC subsystem processes system commands that you issue to the DBC subsystem by using the GROUP value.
**Note**

A DBC XCF group can contain only one DBC subsystem per LPAR.

The default value is DBCGROUP.

**Data type:** VARCHAR(8)  
**Parent element:** <OPTIONS>  
**Child elements:** None.

**XCFGROUP**

*(optional)* By default, DBC uses the `<GROUP>` element as the name of an XCF group to join at startup. In rare situations, this behavior might not be acceptable. Use the `<XCFGROUP>` element to specify a different XCF group for DBC to join.

**Data type:** VARCHAR(8)  
**Parent element:** <OPTIONS>  
**Child elements:** None.

**MODE**

MODE controls whether to initialize products stored in the repository when starting. Valid values for the `<MODE>` element are SAFE and NORMAL.

- SAFE mode tells the DBC subsystem to start without automatically initializing products stored in the repository.
- NORMAL mode (the default value) initializes products stored in the repository.

You can also specify the value for MODE through start arguments to DBCMAIN.

If DBC starts in SAFE mode, DBC issues the following message to the system log:

```plaintext
BMCDBC0250 ssid DBC has been requested to ignore INITPROD commands from the repository
```

If starting in default NORMAL mode, DBC issues the following message:

```plaintext
BMCDBC0234I DBC operating in NORMAL mode.
```

**Data type:** VARCHAR(6)  
**Parent element:** <OPTIONS>  
**Child elements:** None.

**WLM**

The `<WLM>` element sets the subsystem type to use for workload management. The default is BMCS.
Data type: Not applicable.
Parent element: <OPTIONS>
Child elements: <SUBSYSTYPE>

SUBSYSTYPE

The <SUBSYSTYPE> element contains the name of the subsystem type.

Data type: VARCHAR(8)
Parent element: <OPTIONS>
Child elements: <SUBSYSTYPE>

WTOUPPERCASE

(optional) The <WTOUPPERCASE> element indicates whether to convert console messages to uppercase. Printed messages remain in mixed case.

The default value is YES. Valid values are YES and NO.

Data type: VARCHAR(3)
Parent element: <OPTIONS>
Child elements: None.

DPRREPOS

(optional) The <DPRREPOS> element contains the DPR repository name and allocation options. If you do not specify this element, the repository services of the DPR component are unavailable and the DBC issues a warning message during initialization; however, DBC subsystem services function normally with the exception of the <AUTOEXEC> features that require a persistent repository data set.

Data type: Not applicable.
Parent element: <OPTIONS>
Child elements: <NAME>, <STORCLAS>, <VOLUMES>, <MGMTCLAS>, <DATACLAS>, and <IGNORE>

NAME

(optional) The <NAME> element specifies the data set name for the DPR repository VSAM cluster. This value must specify a valid data set name:

- If the value specifies the name of an existing DPR repository VSAM cluster, DPR repository services use that cluster for the repository.

- If the specified data set does not exist, DPR repository services define the new cluster automatically by using the <STORCLAS> or <VOLUMES> options to determine physical allocation attributes.
If you do not provide a value in the <NAME> element, DBC issues a warning message during initialization, and the repository services are unavailable.

Only one active DBC subsystem can use a given DPR repository VSAM cluster.

You can share a single DBCPARMS parameters file for multiple DBC instances. To do so, specify DBC system variables within the <NAME> element to ensure that each DBC subsystem uses a unique repository data set name. DBC resolves the variables to their symbolic values. To enable variable substitution, prefix the system variables with an ampersand (&) and suffix them with a period (.)

Example
Assume that you use the system variables in a <NAME> element as shown in the following statement:

```
<NAME>BMC.&SSID..&SMFID..&MVSNAME..REPOS</NAME>
```

Also, assume that the symbolic values of SSID, SMFID, and MVSNAME are DBC1, SYSP, and MVSPROD, respectively.

The variable substitution resolves to the following data set name:

```
BMC.DBC1.SYSP.MVSPROD.REPOS
```

Data type: VARCHAR(44)
Parent element: <DPRREPOS>
Child elements: None.

STORCLAS

(optional) The <STORCLAS> element specifies the SMS storage class to be used for the DPR repository VSAM cluster when the DPR defines the cluster. The DPR defines the cluster automatically if the <NAME> element specifies the name of a cluster that has not yet been defined. Specify this element only if SMS is active and the cluster is to be SMS-managed. For more information, see the STORAGECLASS parameter of the DEFINE CLUSTER command in the IBM DFSMS Access Method Services documentation.

Data type: VARCHAR(8)
Parent element: <DPRREPOS>
Child elements: None.

VOLUMES

(optional) The <VOLUMES> element specifies the volumes on which the DPR repository VSAM cluster is to be defined when the DPR defines the cluster. The DPR defines the cluster automatically if the <NAME> element specifies the name of a cluster that has not yet been defined.
Note
This element is required only if you do not specify if a value for
<VSTORCLAS>.

The <VOLUMES> value can specify up to 59 volumes, where each
volume is delimited by a comma (,). The value can specify a volume
serial number or, if the cluster is to be SMS-managed, an asterisk (*),
which lets SMS choose the volume. For more information, see the
VOLUMES parameter of the DEFINE CLUSTER command in the IBM
DFSMS Access Method Services documentation.

Data type: VARCHAR(412)
Parent element: <DPRREPOS>
Child elements: None.

MGMTCLAS

(optional) The <MGMTCLAS> element specifies the management class
for the repository data set to be created.

The <MGMTCLAS> element is used only if the repository data set does
not yet exist.

Data type: VARCHAR(8)
Parent element: <DPRREPOS>
Child elements: None.

DATAACLAS

(optional) The <DATAACLAS> element specifies the data class for the
repository data set to be created.

The <DATAACLAS> element is used only if the repository data set does
not yet exist.

Data type: VARCHAR(8)
Parent element: <DPRREPOS>
Child elements: None.

IGNORE

(optional) The <IGNORE> element tells DBC to ignore the repository
entries for the identified product at startup. Doing so allows the rest of
the products running within the DBC subsystem to start normally
without starting the specified product. You can include multiple
<IGNORE> elements.

when DBC is started Data type: Not applicable.
Parent element: <DPRREPOS>
Child elements: <PRODUCT>, <FMID>, and <PIID>
PRODUCT

(required) The <PRODUCT> element identifies a unique BMC product code.

Data type: CHAR(3)
Parent element: <IGNORE>
Child elements: None.

FMID

(required) The <FMID> element identifies the FMID of the product code release.

Data type: CHAR(7)
Parent element: <IGNORE>
Child elements: None.

PIID

(optional) The <PIID> element is optional and identifies a unique product instance within the context of one product code and FMID.

Data type: VARCHAR(16)
Parent element: <IGNORE>
Child elements: None.

Stopping the DBC subsystem

Use the following procedure to stop the DBC subsystem through an IBM z/OS console command. You identify the DBC subsystem to stop through the subsystem ID (ssid) or XCF group value (group).

WARNING

Shutting down the DBC subsystem also shuts down all associated Subzero servers used by BMC Subsystem Optimizer for zEnterprise. If a server shuts down while active connections exist between the client and its remote subsystem, abends can occur in the Subzero, client, and subsystem address spaces.

Before you terminate the DBC subsystem or any Subzero address space, execute the processes and procedures that properly terminate a connection between a client and its connected subsystem.

To stop the DBC subsystem

1. Enter one of the following z/OS commands:
   - ssid STOP
   - group STOP
   - MODIFY ssid,STOP
Example
Assume that a DBC subsystem has an SSID of DBC1. You can stop this subsystem by issuing the STOP command as a MODIFY command:
```
F DBC1,STOP
```

**WARNING**
Some product processes might require an extended amount of time to terminate successfully. If DBC shutdown is delayed, review DBCPRINT for the following message to determine the cause of the delay:
```
BMCDBC0232I... DBC is waiting for product productName to shutdown.
```

If the DBC does not shut down after a reasonable amount of time, you can use the following command to terminate the delayed process:
```
ssid STOP,FORCE
```

Use caution when doing so; terminating a process that is still in use can have unpredictable results. BMC recommends that you use the STOP,FORCE command only if preceding attempts to stop the DBC have failed.

**DBC commands**

You can interact with the DBC subsystem by using any of the following commands.

The `ssid` variable refers to the subsystem ID of the DBC subsystem.

### Table 46: DBC commands

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ssid DBC,DIAG</code></td>
<td>Sends diagnostic information to the DBCPRINT and DBCTRACE data sets. BMC Support might direct you to execute this command to assist in diagnosing problems.</td>
</tr>
<tr>
<td><code>/ssid DBC,DBC2,ssid,command</code></td>
<td>Executes a DB2 command in the DB2 subsystem identified by &quot;ssid&quot; and returns the result to DBCPRINT. This command can be used to execute any native DB2 command that the user is authorized for. <strong>Note:</strong> This command cannot be used for DSN subcommands.</td>
</tr>
<tr>
<td><code>/ssid DBC,RESETPRINT</code></td>
<td>Closes the DBCPRINT data set and allocates a new one in order to clear data from DBCPRINT</td>
</tr>
<tr>
<td><code>/ssid DBC,RESETTRACE</code></td>
<td>Clears the DBCTRACE data set and allocates a new one</td>
</tr>
<tr>
<td><code>/ssid STOP</code></td>
<td>Stops the DBC subsystem</td>
</tr>
<tr>
<td><code>/ssid DBC,TRACEOFF</code></td>
<td>Disables DBC tracing</td>
</tr>
</tbody>
</table>
### Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ssid DBC,TRACEON</code></td>
<td>Enables DBC tracing. DBC writes trace data to the DBCTRACE data set. The DBCTRACE data set is dynamically allocated. <strong>WARNING</strong>: If you leave tracing enabled, be aware that the DBCTRACE data set can grow to a significant size. BMC recommends using caution when using the TRACEON command.</td>
</tr>
<tr>
<td><code>/ssid DBC,TRACEFLUSH</code></td>
<td>Forc...wr...trace information to the DBCTRACE data set. This command is useful if trace was disabled. DBC writes any trace data that exists in an internal buffer that is used to maintain trace information.</td>
</tr>
<tr>
<td><code>/ssid DBC,PRODCMDS,ppp</code></td>
<td>Displays all commands for product <code>ppp</code> <em>(Displays only commands that conform to the <code>ppp</code>,command format)</em>.</td>
</tr>
<tr>
<td><code>/ssid DBC,DISPLAY,object</code></td>
<td>The details of the DBC, DISPLAY command for each object type are documented in the following table.</td>
</tr>
</tbody>
</table>

### Table 47: DISPLAY Command Options

<table>
<thead>
<tr>
<th>DISPLAY object (alternative syntax)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCTS (PRODUCTS, PRD, PDXD) [,ppp]</td>
<td>Displays the Product Definition XML Document (PDXD) from the DBC repository. If no parameters are specified, information is displayed for all products. If the three-character product code is specified, DBC displays information for only that product. If multiple PIIDS exist for that product code, they are all displayed.</td>
</tr>
<tr>
<td>CMDS [,ppp]</td>
<td>Displays all operator command definitions. If no parameters are specified, commands for all products are displayed. If the three-character product code is specified, DBC displays only commands for that product.</td>
</tr>
<tr>
<td>EVENT (EV),eventName [subid]</td>
<td>Displays all subscription information the named event. If the <code>subid</code> is specified, only information for the named subscription is displayed.</td>
</tr>
<tr>
<td>CMD,cmdtext</td>
<td>Displays the ACTION XML for the named command.</td>
</tr>
<tr>
<td>DISPLAY object (alternative syntax)</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>AGENTS (AE) [,.ppp]</td>
<td>Displays all agents that are currently executing. This command generates message BMCDBC0126I, which indicates the TCB address of the agent and whether it is a job step TCB. If no parameters are specified, this command displays information for all products. If the three-character product code is specified, displays currently execution agents for only the product specified.</td>
</tr>
<tr>
<td>PROCESSES</td>
<td>Displays all active PROCESSES. If no other parameters are specified, PROCESSES for all products are displayed. If the three-character product code is specified, displays currently active PROCESSES for only the product specified.</td>
</tr>
<tr>
<td>DB2 [.ssid][,LOCAL</td>
<td>ALL ]</td>
</tr>
<tr>
<td>IMS [,.imsid][,LOCAL</td>
<td>ALL ]</td>
</tr>
</tbody>
</table>
DISPLAY object (alternative syntax) | Description
--- | ---
MQ [,ssid][,LOCAL | ALL ] | Displays the MQINFO block for Websphere MQ subsystems. If no parameters are specified, this command displays MQ blocks for all subsystems known to the DBC group.

If the ssid is specified, MQINFO blocks for only matching MQ subsystems are displayed. You can partially use wildcards for this value by specifying an asterisk at the end (e.g. MX* selects all MQ subsystems with SSID beginning “MX”).

If neither ALL or LOCAL is specified, the command considers MQ subsystems from all LPARs known to the DBC group. LOCAL causes only subsystems executing on the local LPAR to be selected.

SRBS [,ppp ] | Displays currently active SRBs. If no parameters are specified, SRBs for all products are displayed.

If the three-character product code is specified, displays currently active SRBs only for the product specified.

STATS [,TCBS, name] [,MEM][,QUEUES] | If no parameters are specified, this command displays all available statistics.

name displays statistics for only the named task
MEM displays memory usage statistics
QUEUES displays queue usage statistics

DSNS | Displays all data sets currently allocated and specifies if the data set allocation results from STEPLIB, product, or agent definition.

SOCKETS | Displays the state of all sockets currently in use.

DBC security

This section introduces security for DBC.

Consider the following requirements and features of DBC security:

- DBC must meet the following UNIX requirements:
  
  — Write and execute access to the /tmp directory.
  
  — Update access to the FSACCESS (UNIX file system access check) resource class.
DBC must be authorized to create an Extended MCS Console.

DBC uses the standard System Authorization Facility (SAF) interface to communicate with an External Security Manager (ESM). DBC is compatible with ESMs that support the SAF interface (including CA Technologies CA-ACF2 and CA-Top Secret products).

The DBC security interface is compatible with the IBM Resource Access Control Facility (RACF) Version 1.9 or later. DBC issues security calls directly to the SAF interface.

For more information, see “Configuring RACF security for the DBC subsystem” on page 322.

DBC does not require you to define resources (that is, internal control points) to an ESM. However, if the security parameter <ALLOW_SAF_RC4> is set to NO and no resource names have been defined, all DBC requests fail with an authorization error. If <ALLOW_SAF_RC4> is set to NO, you must define DBC resource names so that access can be granted or denied as appropriate.

DBC security control points include commands that are issued from an IBM z/OS system console and from the IBM System Display and Search Facility (SDSF). The DBC subsystem command processor extracts the user ID from the ACEE that is associated with the console address space and propagates this value through the system. Appropriate ESM customization is required to allow operator authorization to DBC commands.

DBC stores its security parameters in the DBCSECUR data set.

DBC security parameters are independent of all other DBC parameters. This physical separation allows the security administrator to implement independent and discrete access to the security parameters (DBCSECUR) and potentially more general access to the subsystem parameters (DBCPARMS).

For more information about security parameters, see “DBC security parameters” on page 325.

Controlling access to DBC resources

The DBC subsystem always issues a Security Access Facility (SAF) security call for internal resource names. By default, if those resource names are not defined to an ESM, access to undefined DBC resources (internal functional control points) is granted for all users who can communicate to the DBC service address space. This behavior occurs because the value of the DBC <ALLOW_SAF_RC4> security parameter defaults to YES.

If you want to restrict access to DBC services that have not been defined as a resource to SAF, you must set the value of the <ALLOW_SAF_RC4> security parameter to NO. Doing so denies access to all undefined resources.
Figure 91 on page 322 shows an example of the ALLOW_SAF_RC4 value.

**Figure 91: Example of the ALLOW_SAF_RC4 value**

```
<DBCSECUR>
  <RESOURCE_NAME>...
  <ALLOW_SAF_RC4>NO</ALLOW_SAF_RC4>
</DBCSECUR>
```

You control this value through DBC security parameters defined in a file identified through the DBCSECUR DD statement in the JCL procedure for the DBC started task.

- To specify the location of the DBC security parameters, see “DBC startup parameters” on page 307.
- To change the value of the `<ALLOW_SAF_RC4>` security parameter, see “DBC security parameters” on page 325.

### Configuring RACF security for the DBC subsystem

The security administrator for a site should perform the following tasks to authorize the DBC subsystem:

- “Authorizing the DBC started task procedure” on page 322
- “Authorizing a user with access to the DBC services” on page 323
- “Creating a SAF resource class (optional)” on page 323
- “Defining a DBC control point resource profile” on page 324

#### Authorizing the DBC started task procedure

The DBC subsystem acquires security authorization from the USERID associated with the started task procedure that starts the DBC subsystem. Users must authorize the DBC started task.

**To authorize the DBC started task**

1. Associate the DBC started task with a USERID that has the appropriate security access.

   The DBC address space requires access to z/OS UNIX System Services (USS). Consequently, you must associate an OMVS segment with the USERID for the DBC subsystem.
Example
The following example uses a RACF command to define a GROUP of DBCGRP:
ADDUSER DBCGRP OMVS(GID(groupIdentifier))

The following example uses a RACF command to define a USERID of DBCUSR:
ADDUSER DBCUSR DFLTGRP(SYSMGMT) OWNER(SYSPROG)

2 Associate the USERID with the procedure name for the DBC started task.

Example
In the following example, the RACF command uses procedure name DBCSTC:
RDEFINE STARTED DBCSTC.* STDATA(USER(DBCUSR) GROUP(DBCGRP))

3 To ensure that RACF recognizes these security updates, issue the following command:
SETROPTS RACLIST(STARTED) REFRESH

Authorizing a user with access to the DBC services

To access DBC services, batch and TSO users require access to z/OS UNIX System Services (USS) and must be associated with an OMVS segment.

Example
The following example uses a RACF command to modify the appropriate group for a user:
ALTGROUP groupName OMVS(GID(groupIdentifier))

The following example uses a RACF command to modify the user’s profile:
ALTUSER userName OMVS(UID(userIdentifier))

Creating a SAF resource class (optional)

This section explains how to create a new RACF resource class. Performing this task is not necessary if the predefined RACF FACILITY class is appropriate for all DBC resource definitions.

Note
This section provides general instructions. Your site might have additional considerations. BMC recommends consulting your security administrator when creating your SAF resource class.

To update the RACF resource class descriptor table (ICHRRCDE)

1 Code the ICHERCDE macro for each required resource class.
TITLE 'RACF RESOURCE CLASS DESCRIPTOR TABLE - ICHRRCDE'

* class ICHERCDE CLASS=class. X
2 Assemble the source and link edit the resulting object module to replace the current ICHRRCDE load module.

To update the RACF router table (ICHRFR01)

1 Code the ICHRFRTB macro for each required resource class.

```
TITLE 'RACF ROUTER TABLE - ICHRFR01'
*
class ICHRFRTB CLASS=class.
ACTION=RACF
*
  ICHRFRTB TYPE=END
* 
END
```

2 Assemble the source and link edit the resulting object module to replace the current ICHRFR01 load module.

3 If necessary to activate the RACF table changes, perform an IPL of the system.

4 Issue the following RACF command for each new resource class:

```
SETROPTS CLASSACT(class)
```

5 (optional) If you use generic profiles, issue the following RACF commands for each new resource class:

```
SETROPTS GENERIC(class)
SETROPTS GENCMD(class)
```

Defining a DBC control point resource profile

The DBC subsystem and its components automatically call the SAF router to check user authorization to various services.

These services are identified by internal functional control points and are externally associated with a resource name. You control user access to the DBC component services by granting or denying authorization to the resource names that are associated with these internal functional control points. To control access to these services, you must define these resource names to the ESM.
To define a DBC control point resource profile

1 Define the resource profile (that is, the resource name) to the RACF ESM by using one or more RDEFINE FACILITY commands.

Example

The following example protects access to various resource categories for a DBC subsystem. The profile applies to LPARs named PROD. You set the PROD value through the < CONTEXT> XML element in the DBC security parameters. The profile also applies to product codes DBC and DPR (inherent components of the DBC subsystem) and ABC (which relates to a DPR-initialized product with the 3-byte product code ABC). For example, the BMC System and SQL Performance products for DB2 use DBC, DPR, LGC, and NGL.

RDEFINE FACILITY (BMC.DBC.PROD.*) UACC(NONE)
RDEFINE FACILITY (BMC.DPR.PROD.*) UACC(NONE)
RDEFINE FACILITY (BMC.ABC.PROD.*) UACC(NONE)

The next example defines a generic profile that protects all currently defined subsystem resources and future resources that are associated with products that you have not yet defined to the DPR component of DBC:

RDEFINE FACILITY (BMC.*.PROD.*) UACC(NONE)

2 Activate the resource class by issuing one of the following commands:

- SETROPTS CLASSACT(FACILITY)
- SETROPTS CLASSACT(FACILTY) RACLIST(FACILITY) (to maintain profiles in memory)

3 (optional) Enable generic profile checking for the FACILITY class:

SETROPTS GENERIC(FACILITY)

DBC security parameters

The DBCSECUR DD statement, which is specified in the startup JCL for the DBC subsystem, identifies the security parameters data set for the DBC subsystem. The security parameters are optional. Each security parameter has a default value that applies if you do not specify the DBCSECUR DD statement, or if you omit a particular security option from the parameters file. Thus, you are not required to specify these parameters to use the DBC security features.

You do not specify the security parameters with the main DBC startup parameters identified in the DBCPARMS DD statement. You can choose to administer and secure the security parameters separately from the main DBC startup parameters by implementing RACF data set name security. For more information about the started task, see “Started task for the DBC subsystem” on page 305.
Structure of the XML stream

Figure 92 on page 326 shows a sample security parameters file.

Figure 92: Structure of the DBCSECUR XML stream

```
<DBCSECUR>
  <RESOURCE_NAME>
    <HLQ>BMC</HLQ>
    <CONTEXTS>
      <CONTEXT>
        <SMFID>MVSA</SMFID>
        <TO_VALUE>PROD</TO_VALUE>
      </CONTEXT>
      <CONTEXT>
        <SMFID>MVSB</SMFID>
        <TO_VALUE>TEST</TO_VALUE>
      </CONTEXT>
    </CONTEXTS>
  </RESOURCE_NAME>
  <RESOURCE_CLASS>
    <COMPONENT>DBC
      <COMMAND>MYCLASS</COMMAND>
    </COMPONENT>
    <COMPONENT>DPR
      <COMMAND>MYCLASS</COMMAND>
    </COMPONENT>
  </RESOURCE_CLASS>
  <SUBSYS>DBCS</SUBSYS>
  <ALLOW_SAF_RC4>NO</ALLOW_SAF_RC4>
</DBCSECUR>
```

DBCSECUR elements

*(optional)* The `<DBCSECUR>` element is the root-level element of the DBCSECUR structure.

*Data type:* Not applicable.
*Child elements:* `<RESOURCE_NAME>`, `<RESOURCE_CLASS>`, `<SUBSYS>`, `<ALLOW_SAF_RC4>`, and `<DB2AUTH>`

**RESOURCE_NAME**

*(optional)* The `<RESOURCE_NAME>` element contains the options for the customizable resource name nodes.

*Data type:* Not applicable.
*Parent element:* `<DBCSECUR>`
*Child elements:* `<HLQ>` and `<CONTEXTS>`

**HLQ**

*(optional)* The `<HLQ>` element specifies a value for the HLQ node of the resource name structure. This value defaults to BMC.

*Data type:* VARCHAR(8)
*Parent element:* `<RESOURCE_NAME>`
*Child elements:* None.
CONTEXTS

(optional) The <CONTEXTS> element contains one or more context specifications.

Data type: Not applicable.
Parent element: <RESOURCE_NAME>
Child elements: <CONTEXT>

CONTEXT

(required) The <CONTEXT> element specifies a value for the context node of the resource name structure. This option defaults to the SMFID that is associated with the IBM z/OS image on which the DBC subsystem is active. The <CONTEXT> element must specify a <SMFID> and <TO_VALUE> element.

Data type: Not applicable.
Parent element: <RESOURCE_NAME>
Child elements: <SMFID> and <TO_VALUE>

SMFID

(required) The <SMFID> element specifies the SMFID that is associated with the z/OS image on which the DBC subsystem is active.

Data type: VARCHAR(4)
Parent element: <CONTEXT>
Child elements: None.

TO_VALUE

(required) The <TO_VALUE> element specifies the value that the DBC subsystem uses for the context node of the resource name.

Data type: VARCHAR(4)
Parent element: <CONTEXT>
Child elements: None.
The following example tells the DBC subsystem to use PROD as the context node of the resource name if the subsystem is started on the z/OS image with an SMFID of LPR1:

```xml
<CONTEXT>
  <SMFID>LPR1</SMFID>
  <TO_VALUE>PROD</TO_VALUE>
</CONTEXT>
```

If not started on this z/OS image, the DBC subsystem ignores the <CONTEXT> specification. This behavior allows you to define a single SAF options file for multiple DBC subsystems that run on different z/OS images.

By modifying the <CONTEXT> element to point a specific LPAR to a logical name, you can define RACF resource names that are consistent across multiple LPARs.

**RESOURCE_CLASS**

*(optional)* The `<RESOURCE_CLASS>` element allows you to customize the SAF resource class that is associated with internal DBC security control points. If omitted, the RACF resource class for all DBC commands (and associated components) defaults to the FACILITY class.

**Note**

This value does not affect the SAF resource class for DPR-initialized product objects. You can customize those classes by using the `<SAFCLASS>` XML tag in the product definition XML document.

In the sample shown in Figure 92 on page 326, all DBC and DPR command resource profiles must be defined in RACF resource class MYCLASS.

The `<COMPONENT>` and `<COMMAND>` subelements are required only if you use the `<RESOURCE_CLASS>` element.

**Data type:** Not applicable.

**Parent element:** `<DBCSECUR>`

**Child elements:** `<COMPONENT>`

**COMPONENT**

*(required)* The `<COMPONENT>` element identifies the specific component for which you are defining the resource class.

**Data type:** CHAR(3)

**Parent element:** `<RESOURCE_CLASS>`

**Child elements:** `<COMMAND>`
COMMAND

(required) The <COMMAND> element identifies the SAF resource class name that is used in all SAF security calls for commands issued from the associated component.

Data type: VARCHAR(8)
Parent element: <COMPONENT>
Child elements: None.

SUBSYS

(optional) The <SUBSYS> element specifies the value to be passed to SAF on each authorization check to the SUBSYS parameter on the RACROUTE macro. The value defaults to DBCS.

Data type: VARCHAR(8)
Parent element: <DBCSECUR>
Child elements: None.

--- Note ---

DBC uses BMCDBC as the application name that is passed to SAF through the APPL parameter on the RACROUTE REQUEST=AUTH macro call. This parameter specifies the name of the application that is making the authorization request. The RACROUTE service makes the parameter available to the installation exit routine, or any routines that the service invokes.

ALLOW_SAF_RC4

(optional) The <ALLOW_SAF_RC4> element specifies whether the DBC subsystem allows access to a given resource if SAF returns return code 4. SAF returns 4 if a security decision could not be made.

Valid values are YES and NO:

- YES (the default) tells the DBC subsystem to allow user access to a resource if SAF returns return code 4.

- NO tells the DBC subsystem not to allow user access to the resource.

Data type: VARCHAR(3)
Parent element: <DBCSECUR>
Child elements: None.

--- Note ---

The IBM RACROUTE macro reference documentation (SA22-7692-04) documents the SAF return codes.
OMVS segment requirements and ESM definitions

This appendix provides information about the requirements and usage of OMVS segments by MainView products. This appendix also provides examples of how to define OMVS segments to various external security manager (ESM) products.

General OMVS segment requirements

The first table in this topic lists the OMVS segment requirements sorted by product or component name.

Table 49 on page 335 lists the OMVS segment requirements sorted by the type of address space.

Table 48: OMVS segment requirements by product or component

<table>
<thead>
<tr>
<th>Product or component</th>
<th>Runs in this type of address space</th>
<th>OMVS segment required?</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Discovery for z/OS</td>
<td>PAS</td>
<td>yes</td>
<td>Superuser authority is required; use either UID 0 or BPX.SUPERUSER.</td>
</tr>
<tr>
<td>CMF MONITOR</td>
<td>z/OS PAS</td>
<td>maybe</td>
<td>If CMFCPMxx or CMFIPMxx includes the OMVS extractor statement, to run this control statement, the user ID under which the Extractor is running must be designated as a valid OMVS user ID. You can use the security system facilities for your system to define the user ID to OMVS. If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>Product or component</td>
<td>Runs in this type of address space</td>
<td>OMVS segment required?</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------</td>
<td>------------------------</td>
<td>-------</td>
</tr>
</tbody>
</table>
| MainView AutoOPERATOR | BBI-SS PAS                         | maybe                   | You need to define an OMVS segment for your PAS when one of the following conditions is true:  
  ■ if the PAS uses TCP/IP services (as a TCP/IP server or client)  
  ■ if using TCP/IP in the PAS (for GTS, which includes MainView for MQ and MainView AutoOPERATOR) or GME (for MainView AutoOPERATOR) for either a PATROL Enterprise Manager (PEM) or BMC Impact Integration for z/OS (BiiZ) connection (see “OMVS segment requirements for TCP/IP communication” on page 338)  
  ■ if any EXECs use OMVS services |
<p>| MainView Explorer    | PAS                                | yes                     | none |
| MainView for CICS    | BBI-SS PAS                         | maybe                   | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for DB2     | BBI-SS PAS                         | maybe                   | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for DBCTL   | BBI-SS PAS                         | maybe                   | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for IMS Online | BBI-SS PAS                | maybe                   | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |</p>
<table>
<thead>
<tr>
<th>Product or component</th>
<th>Runs in this type of address space</th>
<th>OMVS segment required?</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MainView for IP</td>
<td>PAS</td>
<td>yes</td>
<td>A security product like Resource Access Control Facility (RACF) is required for the OS/390 eNetwork Communications Server version 2.5 (or newer) IP environment. Each unit of work in the system that requires UNIX System Services must be associated with a UNIX System Services identity. A valid identity refers to the presence of a valid UNIX user ID (UID), a valid UNIX group ID (GID), and a valid HOME directory for the user. The UID and the GID are defined through the OMVS segment, in the RACF profile, and in the RACF group profile. If the started task is not defined to a user ID that has a defined OMVS segment, error messages might be issued. If the HOME directory is not specified for the user ID, the socket call might fail.</td>
</tr>
</tbody>
</table>
| MainView for Java Environments | z/OS PAS | yes | Superuser authority is required. Assign one of the following privileges:  
  - User ID authorized read access to Facility class resource BPX.SUPERUSER  
  - User ID authorized read access to UNIXPRIV class resource SUPERUSER.PROCESS.GETPSENT  
  - UID=0 to the PAS OMVS segment (BMC does not recommend using this method) |
<p>| MainView for Linux - Servers | PAS | yes | none |
| MainView for UNIX System Services | z/OS PAS | yes | Superuser authority is required. Assign UID=0 or a user ID authorized to read access to RACF class resource BPX.SUPERUSER. |
| MainView for VTAM | PAS | yes | none |
| MainView for WebSphere Application Server | PAS | yes | none |
| MainView for MQ | BBI-SS PAS | yes | none |
| MainView for z/OS | z/OS PAS | maybe | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |</p>
<table>
<thead>
<tr>
<th>Product or component</th>
<th>Runs in this type of address space</th>
<th>OMVS segment required?</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MainView Infrastructure</td>
<td>CAS</td>
<td>yes</td>
<td>CAS-to-CAS communication using TCP/IP needs an OMVS segment. MainView Alarm Management version 5.0 or newer needs an OMVS segment.</td>
</tr>
<tr>
<td>MainView SRM</td>
<td>PAS</td>
<td>maybe</td>
<td>The MainView SRM Started Task (SVOS) requires an OMVS segment and UNIX superuser authority for the line command HFU. If your site uses HFS file systems, verify that a UID of 0 (superuser authority) is assigned in the UID parameter in the OMVS segment of the ADDUSER or ALTUSER commands in the RACF profile. For additional information, see the MainView SRM Customization Guide MainView SRM Customization Guide. If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>MainView SYSPROG Services</td>
<td>z/OS PAS</td>
<td>maybe</td>
<td>If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>MainView Transaction Analyzer</td>
<td>PAS</td>
<td>maybe</td>
<td>If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>BMC Runtime Component System (RTCS)</td>
<td>RTCS subsystem STC</td>
<td>no</td>
<td>none</td>
</tr>
<tr>
<td>RTCS</td>
<td>RTCS initiator STC</td>
<td>no</td>
<td>none</td>
</tr>
<tr>
<td>TSO user</td>
<td>UAS</td>
<td>maybe</td>
<td>If TCP/IP is being used for CAS-to-CAS communication, the TSO user's address space (UAS) will need an OMVS segment (see “OMVS segment requirements for TCP/IP communication” on page 338). If default OMVS segments are defined for a system, TCP/IP access does not need OMVS segments to be explicitly defined for the user ID.</td>
</tr>
</tbody>
</table>
### Table 49: OMVS segment requirements by address space

<table>
<thead>
<tr>
<th>Product or component</th>
<th>Runs in this type of address space</th>
<th>OMVS segment required?</th>
<th>Notes</th>
</tr>
</thead>
</table>
| MainView AutoOPERATOR      | BBI-SS PAS                         | maybe                  | You need to define an OMVS segment for your PAS when one of the following conditions is true:  
  ■ if the PAS uses TCP/IP services (as a TCP/IP server or client)  
  ■ if using TCP/IP in the PAS (for GTS, which includes MainView for MQ and MainView AutoOPERATOR) or GME (for MainView AutoOPERATOR) for either a PATROL Enterprise Manager (PEM) or BMC Impact Integration for z/OS (BiiZ) connection (see “OMVS segment requirements for TCP/IP communication” on page 338)  
  ■ if any EXECs use OMVS services  
  See “OMVS segment requirements for TCP/IP communication” on page 338. |
| MainView for CICS           | BBI-SS PAS                         | maybe                  | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for DB2            | BBI-SS PAS                         | maybe                  | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for DBCTL          | BBI-SS PAS                         | maybe                  | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView for IMS            | BBI-SS PAS                         | maybe                  | If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338). |
| MainView MQ                 | BBI-SS PAS                         | yes                    | none                                                                 |
| MainView Infrastructure     | CAS                                | yes                    | CAS-to-CAS communication using TCP/IP needs an OMVS segment.  
  MainView Alarm Management version 5.0 or newer needs an OMVS segment. |
<table>
<thead>
<tr>
<th>Product or component</th>
<th>Runs in this type of address space</th>
<th>OMVS segment required?</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMF MONITOR</td>
<td>z/OS PAS</td>
<td>maybe</td>
<td>If CMFCPMxx or CMFIPMxx includes the OMVS extractor statement, to run this control statement, the user ID under which the Extractor is running must be designated as a valid OMVS user ID. You can use the security system facilities for your system to define the user ID to OMVS. If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>MainView for UNIX System Services</td>
<td>z/OS PAS</td>
<td>yes</td>
<td>Superuser authority is required. Assign UID=0 or a user ID authorized to read access to RACF class resource BPX.SUPERUSER.</td>
</tr>
<tr>
<td>MainView for z/OS</td>
<td>z/OS PAS</td>
<td>maybe</td>
<td>If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>MainView SYSPROG Services</td>
<td>z/OS PAS</td>
<td>maybe</td>
<td>If using TCP/IP for CAS communications, an OMVS segment is required (see “OMVS segment requirements for TCP/IP communication” on page 338).</td>
</tr>
<tr>
<td>BMC Discovery for z/OS</td>
<td>PAS</td>
<td>yes</td>
<td>Superuser authority is required; use either UID 0 or BPX.SUPERUSER.</td>
</tr>
<tr>
<td>MainView Explorer</td>
<td>PAS</td>
<td>yes</td>
<td>none</td>
</tr>
<tr>
<td>MainView for IP</td>
<td>PAS</td>
<td>yes</td>
<td>A security product like Resource Access Control Facility (RACF) is required for the OS/390 eNetwork Communications Server version 2.5 (or newer) IP environment. Each unit of work in the system that requires UNIX System Services must be associated with a UNIX System Services identity. A valid identity refers to the presence of a valid UNIX user ID (UID), a valid UNIX group ID (GID), and a valid HOME directory for the user. The UID and the GID are defined through the OMVS segment, in the RACF profile, and in the RACF group profile. If the started task is not defined to a user ID that has a defined OMVS segment, error messages might be issued. If the HOME directory is not specified for the user ID, the socket call might fail.</td>
</tr>
<tr>
<td>Product or component</td>
<td>Runs in this type of address space</td>
<td>OMVS segment required?</td>
<td>Notes</td>
</tr>
</tbody>
</table>
MainView TCP/IP communication makes use of z/OS UNIX System Services.

A z/OS UNIX security context, called an OMVS segment, is required for any user ID that requests these services.

To use TCP/IP for CAS-to-CAS communication, each of the following types of users must have access to z/OS UNIX System Services:

- started tasks (such as a CAS or PAS)
- TSO user IDs that will be accessing a CAS or PAS (such as the UAS)
- MainView batch report jobs

You can either define a specific OMVS segment for each user ID or use the default OMVS segment that is provided for users and groups. For more information about defining OMVS segments, see the IBM UNIX System Services Planning book.

When you are defining OMVS segments for MainView TCP/IP users, be aware of the following requirements:

- OMVS segments must be defined before you update any CAS definitions to activate TCP/IP communication.
- The PROCUSERMAX value for a PAS must be high enough to support the maximum number of concurrent users in the PAS.
If you do not specify PROCUSERMAX in the OMVS segment for a PAS, the number of processes is limited by the OMVS system parameter MAXPROCUSER. If you assign the same UID to multiple PASs (or share a UID by default), the MAXPROCUSER value applies to the total number of processes in all address spaces that share the same UID.

The MAXPROCSYS value must be high enough to support the increased number of OMVS processes that results from using MainView TCP/IP communication.

The default value specified for MAXPROCSYS in BPXPRMXX is 900. MainView requires 1 process for each connection plus approximately 10 processes for each PAS to support the permanent tasks that run in each address space.

To display the current MAXPROCSYS value and the high-water mark for OMVS processes, issue the following system command:

```
DISPLAY OMVS,L
```

To change the MAXPROCSYS (or MAXPROCUSER) value, use the SETOMVS system command.

After you have defined the necessary OMVS segments and updated the CAS definition with TCP/IP information, you must stop and restart the CAS for the changes to take effect.

When a CAS or PAS that is using TCP/IP communication initializes, a message like the following is displayed:

```
IEF695I START MV42CAST WITH JOBNAME MV42CAST IS ASSIGNED TO USER MVSSTC, GROUP MVSDEV
```

In this case, an OMVS segment was defined for the assigned user name, MVSSTC.

---

### Defining an OMVS segment to security

The topics in this section illustrate how to define an OMVS segment in various ESMs.
RACF

The following example shows how to define an OMVS segment in RACF:

```
ADDUSER MVPAS -
  NAME('BMC Product Name') -
  NOPASSWORD -
  DFLTGRP(SYS1) -
  UACC(READ) -
  LANGUAGE(PRIMARY(ENU)) -
  OMVS( UID(13827) HOME('/home') PROGRAM('/bin/sh') )
```

CA ACF2

Perform the following tasks to define an OMVS segment in CA ACF2:

1. After the LOGONID for the product address space has been inserted, it must have a DIV(OMVS) USER PROFILE that specifies the UID and a DIV(OMVS) GROUP PROFILE that specifies a GID INSERT. For example:

```
SET LID
INSERT USING(ACFSTCID) MVPAS -
  STC NOJOB NOTSO -
  NAME('BMC Product Name') -
  RESTRICT -
  GROUP(SYS1)
SET PROFILE(USER) DIV(OMVS)
INSERT MVPAS UID(17827) HOME('/') OMVSPGM('/BIN/SH')
SET PROFILE(GROUP) DIV(OMVS)
INSERT MVPAS GID(17827)
```

2. After the above statements have been inserted, the following command needs to be entered at the operator console:

```
F ACF2,REBUILD(USR),CLASS(P)
```

CA Top Secret (TSS)

After the ACID for the product address space has been created, the UID, HOME, and OMVSPGM attributes must be defined. For example:

```
TSS CREATE(MVPAS) +
  NAME('BMC Product Name') +
  FACILITY(STC) +
  TYPE(USER) +
  PASSWORD(NOPW,0) +
  MASTFAC(BBI3) +
  DEPT(stcprocs)
TSS ADDTO(MVPAS) +
  UID(17827) HOME('/') +
  OMVSPGM('/BIN/SH') +
  DFLTGRP(OMVSGRP) +
  GROUP(OMVSGRP)
```
For TCP/IP access, only the UID in the user ID OMVS segment and GID in the group OMVS segment are needed. The shell program name and root directory are not needed.

In addition, if default OMVS segments are defined for the system, TCP/IP access does not need OMVS segments for the user ID.
Suppressing U4095 abends

MainView Infrastructure address spaces routinely use U4095 abends to terminate tasks. In most cases, MainView Infrastructure ESTAE processing suppresses the messages and dump that are associated with the abend.

However, there are times when tasks that are running under IBM UNIX System Services must be terminated. If UNIX System Services has an ESTAE set, that ESTAE attempts to force a dump when the U4095 abend occurs. The only way to prevent the dump from being taken is to use a SLIP command.

BBSAMP member BBMSLPBM contains the following SLIP command for suppressing U4095 abends:

```
SLIP SET,EN,ID=XBMC,C=U4095,RE=BADX,A=NODUMP,END
```

You can use this SLIP command in one of two ways:

- Copy BBMSLPBM to SYS1.PARMLIB(IEASLPBM) and add the statement SLIP=BM to an IEACMDxx member.
- Copy the SLIP command to a COMMNDxx or IEACMDxx member.
A

ACID transaction

ACID is an acronym for Atomic, Consistent, Isolation, and Durable.

Atomic refers to a simple model of success; either all changes are committed successfully or none of them are.

Consistent refers to a consistent view of the data. Any inconsistencies will be eliminated by the time the transaction ends and are invisible outside of the transaction.

Isolation refers to the fact that the effects of concurrently running transactions are invisible to one another until the transactions are committed.

The effects of durable transactions persist even in the event of subsequent system failures.

You can consider ACID transaction as a synonym for complex transaction on page 351.

action

A defined operation, such as modifying a MainView window, that is performed in response to a command. See object on page 371.

active window

Any MainView window in which data can be refreshed. See alternate window on page 346, current window on page 353, window on page 385.

administrative view

A display from which a product’s management tasks are performed, such as the DSLIST view for managing historical data sets. See also view on page 384.
aggregate

See zFS aggregate on page 387.

ALT WIN field

An input area where you specify the identifier for an alternative window where the results of a hyperlink are to be displayed. See also alternate window on page 346.

alternate window

(1) An area of the MainView screen that is specifically selected to display the results of a hyperlink. (2) A window whose identifier is defined to the ALT WIN field. See also active window on page 345, current window on page 353, window on page 385, ALT WIN field on page 346.

Alternate Access

See MainView Alternate Access on page 364.

alternate form

A view, requested through the FORM command, that changes the format of a previously displayed view to show related information. See also form on page 358, query on page 375.

analyzer

An online display that presents a snapshot of status and activity data and indicates problem areas. See also CMF MONITOR Analyzer on page 349.

application

(1) A program that performs a specific set of tasks within a MainView product. (2) In MainView VistaPoint, a combination of workloads that display transaction-performance data in a single view.

application trace

See trace on page 381.
ASCH workload

A group of units of work that contain Advanced Program-to-Program Communication (APPC) address spaces.

automatic screen update

A usage mode wherein the currently displayed screen is refreshed automatically at an interval that you specify. This mode is invoked by the ASU command.

B

batch workload

A group of units of work consisting of address spaces that are running accumulated jobs in a single process.

BBI

The basic architecture that distributes work between workstations and multiple IBM z/OS targets for BMC Software MainView products.

BBI-SS PAS

See product address space (PAS) on page 374.

BBPARM

See parameter library on page 372.

BBPROC

See procedure library on page 374.

BBPROF

See profile library on page 374.
BBSAMP

See sample library on page 376.

BBV

See MainView Alternate Access on page 364.

BBXS

The BMC Subsystem Services. A common set of service routines loaded into common storage and used by several BMC Software MainView products.

border

A visual indication of the boundaries of a window.

bottleneck analysis

The process of determining which resources have insufficient capacity to provide acceptable service levels and can, therefore, cause performance problems.

C

CA-Disk

A data-management system by Computer Associates that replaced the DMS product.

CAS

See coordinating address space (CAS) on page 353.

CFMON

See coupling facility monitoring (CFMON) on page 353.

chart

A format for displaying graphical data. See also graph on page 358.
CICSplex

A user-defined set of one or more CICS systems that are controlled and managed as a single functional entity.

CMF MONITOR

See Comprehensive Management Facility MONITOR (CMF MONITOR) on page 352.

CMF MONITOR Analyzer

A batch component of CMF MONITOR that reads the SMF user records and 70-series records created by the CMF MONITOR Extractor and the RMF Extractor and formats the records into printed system-performance reports.

CMF MONITOR Extractor

A component of CMF MONITOR that collects performance statistics for CMF MONITOR Analyzer, CMF MONITOR Online, MainView for z/OS, and RMF postprocessor. See also CMF MONITOR Analyzer on page 349, CMF MONITOR Online on page 349, MainView for z/OS on page 365.

CMF MONITOR Online

A component of CMF MONITOR that uses the MainView window interface to present data about all address spaces, their use of various system resources, and the delays that each address space has incurred while waiting for access to these resources. See also CMF MONITOR on page 349, MainView for z/OS on page 365.

CMF Type 79 API

An application programming interface, provided by CMF MONITOR, that provides access to MainView SMF-type 79 records.

CMFMON

A component of CMF MONITOR that simplifies online retrieval of information about system hardware and application performance and creates MainView SMF-type 79 records.

The CMFMON online facility can be used to view data in one or more formatted screens.
The CMFMON write facility can be used to write collected data as MainView SMF-type 79 records to an SMF or sequential data set.

CMRDETL

A MainView for CICS data set that stores detail transaction records (type 6E) and abend records (type 6D). Detail records are logged for each successful transaction. Abend records are written when an abend occurs. Both records have the same format when stored on CMRDETL.

CMRSTATS

A MainView for CICS data set that stores CICS operational statistic records (at five-minute intervals) and other records (at intervals defined by parameters that are specified during customization by using CMRSOPT).

collection interval

The length of time that data is gathered. See also delta mode on page 354, total mode on page 381.

column

A vertical component of a view or display—typically containing fields of the same type of information—that varies by the objects associated in each row.

command delimiter

A special character, usually a ; (semicolon), that is used to stack commands that are typed concurrently on the COMMAND line for sequential execution.

COMMAND line

A section in the control area of the display where primary commands can be typed. See also line command column on page 362.

Command MQ for S/390

See MainView for WebSphere MQ (formerly MainView for MQSeries) on page 363.
Command MQ Automation D/S

A collection of Command MQ agents that provide local proactive monitoring for both IBM MQSeries and MSMQM (Microsoft message queue manager). The Command MQ agents operate at the local node level where they continue to perform functions regardless of the availability of the MQM (message queue manager) network. Functionality includes automatic monitoring and restarts of channels, queue managers, queues, and command servers. In cases where automated recovery is not possible, the agents transport critical alert information to a central console.

Command MQ Automation S/390

A Command MQ component that monitors the MQM (message queue manager) networks and intercedes to perform corrective actions when problems arise. Solutions include

- Dead-Letter Queue management
- System Queue Archival
- Service Interval Performance
- Channel Availability

These solutions help ensure immediate relief to some of the most urgent MQM operations and performance problems.

Command MQ for D/S

A product that utilizes a true client/server architecture and employs resident agents to provide configuration, administration, performance monitoring, and operations management for the MQM (message queue manager) network.

COMMON STORAGE MONITOR

A component of MainView for z/OS that monitors usage and reconfigures z/OS common storage blocks.

compatibility mode aggregate

A zFS aggregate that contains only a single zSeries file system. A compatibility mode aggregate is similar to an HFS. See also zFS aggregate on page 387.

complex transaction
A complex transaction consists of two or more components. In other words, a complex transaction is one where the originating component initiates at least one other component.

A transaction might contain more than one component either because the program issued a syncpoint, or because the originating transaction spawned sub-transactions.

For example, a CICS task transaction that links to remote CICS subsystems, makes DB2 SQL calls, uses the IMS/DBCTL interface to access IMS databases, and places messages on WebSphere MQ queues that trigger other CICS transactions. Each of these units of work is a component of the same transaction.

The process of identifying related components is called *correlation*.

**component**

A component in MainView Transaction Analyzer is a unit of work (UOW) in any of the following transactional environments:

- CICS Transaction Gateway
- CICS Transaction Server
- DB2
- IMS
- WebSphere MQ

**composite workload**

A collection of WLM and other groups of units of work. See also constituent workload on page 352.

**Comprehensive Management Facility MONITOR (CMF MONITOR)**

A product that measures and reports on all critical system resources, such as CPU, channel, and device usage; memory, paging, and swapping activity; and workload performance.

**constituent workload**

A member of a composite workload. Constituent workloads in a composite workload usually belong to a single workload class, but sometimes are mixed.
contention

A situation that occurs when the requests for service outnumber the available servers.

context

In a Plex Manager view, the field that contains the name of a target or group of targets specified with the CONTEXT command. See also scope on page 377, service point on page 377, SSI context on page 379, target context on page 380.

CONTEXT command

To specify either a MainView product and a specific target for that product (see target context on page 380) or a MainView product and a name representing one or more targets for that product (see SSI context on page 379).

control statement

(1) An instruction that interrupts a sequence of instructions and transfers control to another part of the program. (2) An instruction that names samplers and other parameters that configure the MainView components to perform specified functions. (3) In CMF MONITOR, the instruction in a parameter library member that is used to identify a sampler in the extractor or a report in the analyzer, or to describe either component’s processing requirements to the operating system.

coordinating address space (CAS)

Address space that is used by the MainView windows environment. The CAS supplies common services and enables communication between linked systems. Each z/OS image requires a separate CAS. Cross-system communication is established through the CAS by using IBM VTAM and XCF communication links.

coupling facility monitoring (CFMON)

Coupling facility views that monitor the activity of your system’s coupling facilities.

current window

In the MainView windows environment, the area where the main dialog with the application takes place. When no window number is specified, the current window is used as the default window destination for commands that are issued on the COMMAND line. See also active window on page 345, alternate window on page 346, window on page 385.
current data

The information that reflects the system in its present state. The two types of current data are real-time data and interval data. See also historical data on page 359, interval data on page 360, real-time data on page 375.

D

DASD

See direct access storage device (DASD) on page 355.

data collector

A program that belongs to a MainView product and that gathers information from various sources and stores the data in records used by views. For example, MainView for z/OS data collectors obtain data from z/OS services, z/OS control blocks, CMF MONITOR Extractor control blocks, and other sources. See also extractor on page 356.

Data Facility Storage Management System (DFSMS)

Data-management, backup, and HSM software from IBM for z/OS mainframes.

Data Set Optimizer (DSO)

A CMF MONITOR Extractor component that uses CMF MONITOR Extractor data to produce reports specifying the optimal ordering of data sets on moveable head devices.

---

**Note**

HEADMOVE is not supported by version 5.8 and later of CMF MONITOR.

---

delta mode

(1) In MainView for DB2 analyzer displays, the difference between the value sampled at the start of the current statistics interval and the value sampled by the current analyzer request. See also statistics interval on page 379. (2) In CMFMON, a mode where certain columns of data reflect the difference in values between one sample cycle and the next cycle. Invoked by the DELta ON command. See also collection interval on page 350, sample cycle on page 376, total mode on page 381.
DFSMS

See Data Facility Storage Management System (DFSMS) on page 354.

direct access storage device (DASD)

(1) A device with rotating recording surfaces that provides immediate access to stored data. (2) Any device that responds to a DASD program.

display

A presentation of data in full-screen mode.

DMR

See MainView for DB2 on page 364.

DMS

(Data Management System) See CA-Disk on page 348.

DMS2HSM

See MainView SRM DMS2HSM on page 367.

DSO

See Data Set Optimizer (DSO) on page 354.

E

EasyHSM

See MainView SRM Reporting on page 367.

EasyPOOL

See MainView SRM Allocation on page 366.
EasySMS

See MainView SRM Allocation on page 366.

element

(1) A data component of a data collector record, shown in a view as a field. (2) In a view, an internal value of a field that is used in product functions.

element Help

Online information about a field in a view. The preferred term is field Help.

Enterprise Storage Automation

See MainView SRM Automation on page 366.

event

A message issued by MainView SRM Automation. User-defined storage occurrences generate events in the form of messages. These events provide an early warning system for storage problems and are routed to user-specified destinations for central viewing and management.

Event Collector

A component for MainView for IMS Online, MainView for IMS Offline, and MainView for DBCTL that gathers data about events in the IMS environment. This data is required for Workload Monitor and optional for Workload Analyzer (except for the workload trace service). This data also is recorded as transaction records (X’FA’) and program records (X’F9’) on the IMS system log for later use by the MainView for IMS Offline components: Performance Reporter and Transaction Accountant.

expand

A predefined link from one display to a related display. See also hyperlink on page 359.

extractor

A program that collects data from services, control blocks, and other sources and keeps the data control blocks to be written as records. See also data collector on page 354.
extractor interval

See collection interval on page 350.

F

fast path

A predefined link between one screen and another screen. To use the fast path, place the cursor on a single value in a field and press Enter. The resulting screen displays more detailed information about the selected value. See also hyperlink on page 359.

field

A group of character positions within a screen or report where you can type or display specific information.

field Help

Online information that describes the purpose or contents of an area on a screen. To display field Help, place the cursor anywhere in a field and press PF1 (HELP). In some products, field Help is accessible from the screen Help that is displayed when you press PF1.

filter

Selection criteria that limits the number of rows that are displayed in a view. Data that does not meet the selection criteria is not displayed. A filter consists of an element, an operator, and an operand (a number or character string). Filters can be implemented in view customization, through the PARm/QPARm commands, or through the Where/QWhere commands. Filters are established against elements of data.

fire

This term indicates that an event has triggered an action. In MainView AutoOPERATOR, when rule-selection criteria matches an incoming event and fires, the user-specified automation actions are performed. This process is also called handling the event.

fixed field

A specific data area that remains stationary at the left margin of a screen when the screen is scrolled either to the right or left.
flow chart

Flow charts are used by MainView Transaction Analyzer to map the flow between components in a complex transaction. For more information, see MainView Explorer flow chart on page 365 or use the MainView Explorer online Help.

form

One of two constituent parts of a view; the other part is query. A form defines how the data is presented; a query identifies the data required for the view. See also query on page 375, view on page 384.

full-screen mode

A presentation of a MainView product application or service on the entire screen with no window information line. See also windows mode on page 385.

G

global command

Any MainView windows environment instruction that can affect all windows in a MainView window area.

graph

A pictorial presentation of data that you select from a MainView window environment view. See also chart on page 348.

H

HFS

See hierarchical file system (HFS) on page 358.

hierarchical file system (HFS)
A type of file organization where all files are members of a directory. Each directory is in turn a member of another directory that is at a higher level in the structure (hierarchy). The highest level of the hierarchy is the root directory. An HFS always has a single file system per data set.

**hierarchical storage management (HSM)**

The automatic movement of files from hard disk to slower, less-expensive storage media. The typical hierarchy is from magnetic disk to optical disk to tape.

**hilevel**

For MainView products, a high-level data set qualifier required by a site’s naming conventions.

**historical data**

(1) Information that reflects the system as it existed at the end of a past recording interval or the duration of several intervals. (2) Any data stored in the historical database and retrieved by using the TIME command. *Contrast with* current data, interval data, and real-time data.

**historical data set**

In MainView products that display historical data, the VSAM cluster file in which data is recorded at regular intervals.

**historical database**

A collection of performance data written at the end of each installation-defined recording interval and containing up to 100 VSAM clusters. Data is extracted from the historical database by using the TIME command. See also *historical data on page 359.*

**HSM**

See *hierarchical storage management (HSM) on page 359.*

**hyperlink**

(1) A preset field in a view or an EXPAND line in a display where you can

- access cursor-sensitive Help
- issue commands
The transfer can be within a single product, to a related view or displayed in a different BMC Software product. Generally, fields that have hyperlinks available are highlighted.

(2) A cursor-activated path from a topic or term in online Help to related information. See also fast path on page 357.

image log

A collection of screen-display records. Image logs can be created for both the BBI-SS PAS and the BBI terminal session. See also Journal log on page 361.

The BBI-SS PAS Image log consists of two alternating data sets: as one data set fills up, the other data set is used. Logging to the BBI-SS PAS Image log stops when both data sets are filled and the first data set is not processed by the archive program.

The terminal session Image log is a single data set that wraps around when full.

IMS Resource Utilization File (IRUF)

A collection of reports that can be either detail (one event, one record) or summarized (more than one event, one record). A detail IRUF is created by processing the IMS system log through a program called IMFEDIT. A summarized IRUF is created by processing one or more detail IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program.

IMSPlex System Manager (IPSM)

An MainView for IMS Online and MainView for DBCTL service that provides Single System Image views of resources and bottlenecks for applications across one or more IMS regions and systems.

interval data

Cumulative information that is gathered during a collection period. Intervals usually last from 15 to 30 minutes, depending on how the recording interval is specified during product customization. See also current data on page 354, historical data on page 359, real-time data on page 375.
Note

If a change is made to the workloads, a new interval is started.

InTune

A product that monitors application program performance and provides information that is used to reduce bottlenecks and delays.

IRUF

See IMS Resource Utilization File (IRUF) on page 360.

J

job activity view

A report about address space consumption of resources. See also view on page 384.

journal

A special-purpose data set that stores the chronological records of operator and system actions.

Journal log

A collection of messages. Journal logs are created for both the BBI-SS PAS and the BBI terminal session.

The BBI-SS PAS Journal log consists of two alternating data sets: as one data set fills up, the other data set is used. Logging to the BBI-SS PAS Journal log stops when both data sets are filled and the first data set is not being processed by the archive program.

The terminal session Journal log is a single data set that wraps around when full.

L

line command

An instruction that you type in a specific column of a view or display. Line commands initiate actions that apply to the data in that particular row.
line command column

An instruction input column on the left side of a view or display. See also COMMAND line on page 350.

Log Edit

In the MainView for IMSOffline program named IMFLEDIT, a function that extracts transaction records (X'FA') and program records (X'F9') from the IMS system log. IMFLEDIT also extracts certain records that were recorded on the system log by IMS. IMFLEDIT then formats the records into a file called the IMS Resource Utilization File (IRUF).

logspace

MainView Logger runs as a started task and supports multiple tasks, known as logspaces, within the same address space. Logspace definitions are identified by the LOGSPACE parameter in the LOGGERxx library member.

One logspace is identified as the primary logspace. The primary logspace is shared by MainView Infrastructure and all MainView products. Additional product-specific logspaces might be required by MainView products that have special data requirements. In these instances, after installing the primary logspace, you install a product-specific logspace for each product that requires it. See also MainView Logger on page 363.

M

MainView

The BMC Software integrated systems-management architecture.

MainView Application Program Interface (MVAPI)

A callable interface based on CLIST or REXX that allows MainView AutoOPERATOR EXECs to access view data for MainView monitor products.

MainView AutoOPERATOR

A product that uses tools, techniques, and facilities to automate routine operator tasks and provide online performance monitoring. MainView AutoOPERATOR achieves high availability through error minimization, improved productivity, and problem prediction and prevention.
MainView Explorer

A product that provides access to MainView products from a web browser running under Windows.

MainView for VTAM

A product that displays application-performance data by application, transaction ID, and LU name. This collected data includes connections, response time statistics, application availability, and application throughput.

MainView for WebSphere MQ (formerly MainView for MQSeries)

This product delivers comprehensive capabilities for configuration, administration, performance monitoring, and operations management for an IBM WebSphere MQ (message queue manager) network.

MainView for WebSphere MQ Integrator

A licensed feature of MainView for MQ that provides comprehensive configuration, administration, performance monitoring, and operations-management capabilities for an IBM WebSphere MQ Integrator message broker network.

MainView Logger

MainView Logger is a stand-alone logging facility that provides MainView Infrastructure or any BMC MainView product a way to log important messages or events in a centralized repository. MainView Logger runs as a started task and supports multiple tasks, known as logspaces, within the same address space.

One logspace is identified as the primary logspace. The primary logspace is shared by MainView Infrastructure and all MainView products. Additional product-specific logspaces might be required by MainView products that have special data requirements, such as MainView Transaction Analyzer. In these instances, after installing the primary logspace, you install a product-specific logspace for each product that requires it. See also delta mode on page 354, logspace on page 362.

MainView for WebSphere Application Server (formerly MainView for WebSphere)

A product that provides extensive information for managing the IBM WebSphere Application Server for the z/OS environment. At the user’s option, information is displayed about HTTP servers, WAS plug-ins, or J2EE/CORBA containers. The product also provides JVM profiling capability.
MainView for DB2

A product that provides real-time and historical application-performance analysis and monitoring for DB2 subsystem management.

MainView for IMS Offline

A product with a Performance Reporter component that organizes data and prints reports that are used to analyze IMS performance; and a Transaction Accountant component that produces cost-accounting and user charge-back records and reports.

MainView for IMS Online

A product that provides real-time application-performance analysis and monitoring for IMS management.

MainView for Linux - Servers

A product that monitors the performance of your Linux systems from the MainView windows interface.

MainView Alarm Management

In conjunction with other MainView products, this component notifies you when an exception occurs. MainView Alarm Management is capable of monitoring multiple systems simultaneously, which means that MainView Alarm Management installed on one system keeps track of your entire sysplex. You can then display a single view that shows exceptions for all MainView performance monitors within your z/OS enterprise.

MainView Alternate Access

This product enables MainView products to be used without TSO by providing access through EXCP and VTAM interfaces.

MainView control area

In the MainView windows environment, the first three lines at the top of the view containing the window information line, the COMMAND line, and the SCROLL, CURR WIN, and ALT WIN fields. The control area cannot be customized. Contrast with MainView window area.
MainView display area

See MainView window area on page 368.

MainView Explorer flow chart

The MainView Explorer flow chart depicts the hierarchical relationships between objects. The flow chart displays an array of nodes along with each node’s sub-nodes.

MainView for CICS

A product that provides real-time application-performance analysis and monitoring for CICS system management.

MainView for MQSeries

See MainView for WebSphere MQ (formerly MainView for MQSeries) on page 363.

MainView for OS/390

See MainView for z/OS on page 365. (Prior to version 2.5, this product was known as MainView for MVS. The name changed to MainView for z/OS with version 2.8.00.)

MainView for z/OS

A system-management application that is built upon the MainView window environment architecture. MainView for z/OS uses the windows interface to provide access to system performance data and other necessary functions in the overall management of an enterprise. (Prior to version 2.8.00, this product was known as MainView for OS/390).

MainView for DBCTL

A product that provides real-time application-performance analysis and monitoring for DBCTL management.

MainView for IP

A product that monitors z/OS mission-critical application performance as it relates to TCP/IP stack usage. Collected data includes availability, connections, response times, routers, service levels, storage, traffic, web cache, and so on.
MainView for UNIX System Services

A system-management application for monitoring the performance of the UNIX System Services from a MainView windows interface.

MainView Selection Menu

An ISPF selection panel that provides access to all MainView windows-mode and full-screen mode products.

MainView SRM

See MainView Storage Resource Manager (SRM) on page 368.

MainView SRM Allocation

A component of MainView SRM that

- provides control over data set allocation and enforcement of allocation and naming standards
- operates at the system level to intercept abend conditions or standards violations, thus providing services without any JCL changes
- provides tools that aid in the conversion to IBM DFSMS and enhance the DFSMS environment after conversion

MainView SRM Automation

A component of MainView SRM that delivers powerful event-generation and storage-automation technology across the storage enterprise. When it is used in conjunction with MainView AutoOPERATOR, automated solutions can be used to

- perform pool, volume, application, or data set-level manipulation
- respond to certain conditions
- perform specific requests

MainView SRM EasyHSM

See MainView SRM Allocation on page 366.
MainView SRM EasyPOOL

See MainView SRM Allocation on page 366.

MainView SRM EasySMS

See MainView SRM Allocation on page 366.

MainView SRM Reporting

A component of MainView SRM that monitors and reports on DASD consumption and allows you to dynamically control DASD utilization. Views enable the DASD administrator to review historic DASD usage and control current and future DASD usage. Physical views of storage devices can be supplemented with user-defined applications that allow for budgeting and measurement by logical groups. MainView SRM Reporting also provides online monitoring and reporting to help storage managers use DFHSM efficiently.

MainView SRM SG-Control

See MainView SRM Reporting on page 367.

MainView SRM StopX37/II

See MainView SRM Allocation on page 366.

MainView SRM StorageGUARD

See MainView SRM Reporting on page 367.

MainView SRM DMS2HSM

A product that facilitates the conversion of CA-Disk, formerly known as DMS, to HSM.

MainView SRM Enterprise Storage Automation

See MainView SRM Automation on page 366.

MainView SRM SG-Auto
A product that provides early-warning notification of storage anomalies and automated responses to those anomalies based on conditions in the storage subsystem.

**MainView Storage Resource Manager (SRM)**

A suite of products that assist in all phases of z/OS storage management. MainView SRM consists of products that perform automation, reporting, trend analysis, and error correction for storage management.

**MainView SYSPROG Services**

*See SYSPROG services on page 380.*

**MainView VistaPoint**

A product that provides enterprise-wide views of performance. Application and workload views are available for CICS, DB2, DBCTL, IMS, or z/OS. Data is summarized at the level of detail needed; for example, views can be for a single target, an z/OS image, or an entire enterprise.

**MainView window area**

A portion of the information display that is not the control area and in which views are displayed and windows are opened. It includes all but the first three lines of the information display. See also *MainView control area on page 364.*

**monitor**

An online service that measures resources or workloads at user-defined intervals and issues warnings when user-defined thresholds are exceeded.

**MTALOG**

This log is specific to MainView Transaction Analyzer and contains diagnostic and other information that is logged internally by MainView Transaction Analyzer components.

**Multi-Level Automation (MLA)**

The user-defined, multiple-step process in Enterprise Storage Automation that implements solutions in a tiered approach, where solutions are invoked one after another until the condition is resolved.
MVALARM

See MainView Alarm Management on page 364.

MVAPIS

See MainView Application Program Interface (MVAPI) on page 362.

MVCICS

See MainView for CICS on page 365.

MVDB2

See MainView for DB2 on page 364.

MVDBC

See MainView for DBCTL on page 365.

MVIMS

See MainView for IMS Offline on page 364 or MainView for IMS Online on page 364.

MVIP

See MainView for IP on page 365.

MVLNX

See MainView for Linux - Servers on page 364.

MVMQS

See MainView for WebSphere MQ (formerly MainView for MQSeries) on page 363 or MainView for WebSphere MQ Integrator on page 363.
MVMVS

See MainView for z/OS on page 365.

MVS PAS

Now known as z/OS PAS. See product address space (PAS) on page 374.

MVScope

A MainView for z/OS application that traces CPU usage down to the CSECT level and I/O usage down to the channel-program level.

MVSRM

See MainView Storage Resource Manager (SRM) on page 368.

MVSRMHSM

See MainView SRM Reporting on page 367.

MVSRMSGC

See MainView SRM Reporting on page 367.

MVSRMSGD

See MainView SRM Reporting on page 367.

MVSRMSGP

See MainView SRM Reporting on page 367.

MVUSS

See MainView for UNIX System Services on page 366.

MVVP
See MainView VistaPoint on page 368.

MVVTAM

See MainView for VTAM on page 363.

MVWEB

See MainView for WebSphere Application Server (formerly MainView for WebSphere ) on page 363.

N

nested Help

Multiple layers of Help pop-up windows. Each successive layer is accessed by clicking a hyperlink from the previous layer.

O

object

Anything that you can manipulate as a single unit. A MainView object can be a product, secondary window, view, row, column, or field. You can issue an action against an object by issuing a line command in the line command column to the left of the object. See also action on page 345.

OMVS workload

A group of units of work consisting of z/OS OpenEdition address spaces.

online Help

Explanatory or instructional information that is accessible from within a product.

originating subsystem

The originating subsystem is the transactional environment (CICS, DB2, or IMS) in which the first component of a complex transaction is performed.
If a transaction starts outside the environment that is monitored by MainView Transaction Analyzer, the originating subsystem is that of the first component that actually runs within the monitored environment.

**originating system**

The originating system is the z/OS system image on which the first component of a complex transaction is performed.

If a transaction starts outside the environment that is monitored by MainView Transaction Analyzer, the originating system is that of the first component that actually runs within the monitored environment.

**OS/390 PAS**

Now known as z/OS PAS. See product address space (PAS) on page 374.

**OS/390 and z/OS Installer**

A BMC Software common installation system for mainframe products.

**P**

**parameter library**

A data set consisting of members that contain parameters for specific MainView products or a support component. The following versions can exist:

- the distributed parameter library, called BBPARM
- a site-specific parameter library or libraries

These libraries can be created by customization, and named UBBPARM, or created manually, with a unique name.

**PAS**

See product address space (PAS) on page 374.

**PERFORMANCE MANAGER**
A MainView for CICS online service for monitoring and managing current performance of CICS regions.

**Performance Reporter**

A product component that generates offline batch reports. The following products can generate these reports:

- MainView for DB2
- MainView for CICS
- MainView for IMS Offline

**Performance group (PRGP) workload**

A collection of address spaces defined to z/OS. If you are running z/OS with WLM in compatibility mode, MainView for z/OS creates a performance group workload instead of a service class.

In MVS/SP 5.0 or earlier, or in compatibility mode in MVS/SP 5.1 or later, is a composite of service classes. MainView for z/OS creates a performance group workload for each performance group that is defined in the current IEAIPSxx member.

**Plex Manager**

A product through which cross-system communication, MainView security, and an SSI context are established and controlled. Plex Manager is shipped with MainView windows-environment products as part of the coordinating address space (CAS) and is accessible as an option from the MainView Selection Menu.

**pop-up display**

A full-screen panel that displays additional information about a selected event in a detail trace.

**pop-up window**

Help information in a viewing area that, when active, overlays part of the window area. A pop-up window is displayed when you issue the HELP command while working in windows mode.
procedure library

A data set consisting of members that contain executable procedures that are used by MainView AutoOPERATOR. These procedures are execution command lists (EXECs) that automate site functions. The following versions can exist:

- the distributed parameter library, called BBPROC
- a site-specific parameter library or libraries

These libraries can be created by customization, and named UBBPROC, or created manually, with a unique name.

The site-created EXECs can be either user-written or customized by BBPROC EXECs that are supplied by MainView AutoOPERATOR.

product address space (PAS)

Address space that is used by a MainView product that contains data collectors and other product functions. MainView products use a z/OS PAS, a BBI-SS PAS, or a product-specific PAS.

product-specific PAS

See product address space (PAS) on page 374.

profile library

A data set consisting of members that contain characteristic information and cycle-refresh definitions for a terminal session that is connected to a BBI-SS PAS. Other members are dynamically created by MainView applications. The following versions can exist:

- the distributed profile library, called BBPROF
- a site-specific profile library or libraries

These libraries can be created by customization, and named SBBPROF, or created manually, with a unique name.

The site library is a common profile, shared by all site users. The terminal session CLIST creates a user profile automatically called userid.BBPROF (if a profile does not exist), where userid is your logon ID. User-profile libraries allow each user to specify unique PF keys, CYCLE commands, target system defaults, a Primary Option Menu, and a set of application profiles.
Q

query

One of two constituent parts of a view (the other is form). A query defines the data for a view; a form defines the display format. See also form on page 358, view on page 384.

R

real-time data

Performance information as it exists at the moment of inquiry. Real-time data is recorded during the smallest unit of time for data collection. See also current data on page 354, historical data on page 359, interval data on page 360.

Resource Analyzer

The online real-time displays that are used to examine IMS resources and determine which resources are affected by specific workload problems.

Resource Monitor

The online data collection services that are used to oversee IMS resources and issue warnings when defined utilization thresholds are exceeded.

row

(1) The horizontal component of a view or display comprising all the fields pertaining to a single device, address space, user, and so on. (2) The horizontal component of a DB2 table consisting of a sequence of values (one value for each column of the table).

RTCS

See Runtime Component System (RTCS) on page 375.

Runtime Component System (RTCS)

Address space that provides programming services to all the CASs, PASs, and UASs.
Rx&D2

A product that provides access to DB2 from REXX. Rx&D2 provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more.

S

sample cycle

The time that elapses between data-retrieval points.

For the CMF MONITOR Extractor, this value is the time specified in the extractor control statements (usually 1 to 5 seconds).

For real-time data, the cycle is not fixed. Data is sampled each time you press Enter.

sample library

A data set consisting of members, each of which contains one of the following items:

- sample JCL that can be edited to perform specific functions
- macro that is referenced in the assembly of user-written services
- sample user exit routine

The following versions are available:

- the distributed sample library, called BBSAMP
- a site-specific sample library or libraries

These libraries can be created by customization, and named UBBSAMP, or created manually, with a unique name.

sampler

A program that monitors a specific aspect of system performance. It includes utilization thresholds used by the Exception Monitor. The CMF MONITOR Extractor contains samplers.

SBBPROF
scope

A subset of an SSI context. The scope could be all the data for the context or a subset of data within the context. It is user- or site-defined. See also SSI context on page 379, target on page 380.

screen definition

A configuration of one or more views that have been stored with the SAVEScr command and assigned a unique name. A screen includes the layout of the windows and the view, context, system, and product that are active in each window.

selection view

In MainView products, a list of available views presented within a window.

service point

A specification, to MainView, of the services required to enable a specific product. Services can be actions, selectors, or views. Each target (for example, CICS, DB2, or IMS) has its own service point.

The PLEX view lists all the defined service points known to the CAS to which the terminal session is connected.

service class workload

A collection of address spaces defined to z/OS. If you are running Workload Manager (WLM) in goal mode, MainView for z/OS creates a service class workload for each service class that you define through WLM definition dialogs.

If you are running MVS 4.3 or earlier, or you are running MVS/SP 5.1 or later with WLM in compatibility mode, MainView for z/OS creates a performance group workload instead of a service class. See also performance group (PRGP) workload on page 373.

service objective

A workload performance goal, specified in terms of response time for TSO workloads or turnaround time for batch workloads. Performance group workloads can be measured by either objective. Composite workload service objectives consist of user-defined weighting...
factors assigned to each constituent workload. For compatibility mode, z/OS does not provide any way to measure service.

**service request block (SRB)**

A control block that represents a routine to be dispatched. SRB mode routines generally perform work for the operating system at a high priority. An SRB is similar to a task control block in that it identifies a unit of work to the system. See also task control block (TCB) on page 380.

**service select code**

The code that is entered to invoke analyzers, monitors, and general services. This code is also the name of the individual service.

**session**

The time during which an address space is active. A session begins when monitoring can be performed. If the product address space (PAS) starts after the job, the session starts with the PAS.

**SG-Auto**

See MainView SRM SG-Auto on page 367.

**SG-Control**

See MainView SRM Reporting on page 367.

**single system image (SSI)**

A feature of the MainView windows environment where you can view and perform actions on multiple z/OS systems as though they were a single system. The rows of a single tabular view can contain rows from different z/OS images.

**Skeleton Tailoring Facility**

A facility in MainView AutoOPERATOR that allows JCL that contains variables within the JCL statements to be substituted with data values at job submission time. Directive statements can be used in the skeleton JCL to cause the repetition of a set of skeleton statements. This facility functions similar to the TSO skeleton tailoring facility.
SRB

See service request block (SRB) on page 378.

SSI

See single system image (SSI) on page 378.

SSI context

A name that is created to represent one or more targets for a given product. See also context on page 353, target on page 380.

Started Task workload

The address spaces that are running jobs that were initiated programmatically.

statistics interval

For MainView for DB2, cumulative count within a predefined period of time for an analyzer service DELTA or RATE display. Thirty minutes is the default set by the DB2STATS parameter in the distributed BBPARM member BBISP00. Specifying the DELTA parameter displays the current value as the difference between the value sampled by the current analyzer request and the value sampled at the start of the current interval. Specifying the RATE parameter displays the current value by minute (DELTA divided by the number of elapsed minutes).

stem variables

A REXX facility, supported in MainView AutoOPERATOR REXX EXECs and the Skeleton Tailoring Facility, where variable names end with a period followed by a number, such as &POOL.1. This configuration allows each variable to actually represent a table or array of data, with the zero variable containing the number of entries in the array. For example, &POOL.0 = 5 would indicate that variables &POOL.1 through &POOL.5 exist.

StopX37/II

See MainView SRM Allocation on page 366.

StorageGUARD
See MainView SRM Reporting on page 367.

**summary view**

Customized, formatted data created from a tabular view by using the Summarize option. A summary view compresses several rows of data into a single row based on the summarize criteria.

**SYSPROG services**

A component of MainView for z/OS that offers over 100 functions to detect, diagnose, and correct z/OS system problems as they occur. This component is accessible from the z/OS Performance and Control Main Menu and is also available as a stand-alone product: MainView SYSPROG Services.

**system resource**

See object on page 371.

**T**

**target**

An entity—such as a z/OS image, an IMS or DB2 subsystem, a CICS region, or related workloads across systems—that is monitored by one or more MainView products. See also context on page 353, scope on page 377, SSI context on page 379.

**target context**

A single target/product combination. See also context on page 353.

**TASCOSTR**

A MainView for IMS Offline program that summarizes detail and summary IMS Resource Utilization Files (IRUFs) for use as input to the offline components.

**task control block (TCB)**

An address space-specific control block that represents a unit of work that is dispatched in the address space in which it was created. See also service request block (SRB) on page 378.
TCB

See task control block (TCB) on page 380.

TDIR

See trace log directory (TDIR) on page 382.

terminal session (TS)

A single point of control for MainView products, allowing data manipulation and data display and providing other terminal user services for MainView products. The terminal session runs in a user address space (either a TSO address space or a stand-alone address space for EXCP/VTAM access).

threshold

A specified value that is used to determine whether the data in a field meets specific criteria.

TLDS

See trace log data set (TLDS) on page 381.

total mode

A usage mode in CMFMON wherein certain columns of data reflect the cumulative value between collection intervals. Total mode is invoked by the DELta OFF command. See also collection interval on page 350, delta mode on page 354.

trace

(1) A record of a series of events chronologically listed as they occur. (2) The online data-collection and display services that track transaction activity through DB2, IMS, or CICS.

trace log data set (TLDS)

Single or multiple external VSAM collections of data that contain summary or detail trace data for later viewing or printing. The trace logs can be defined as needed or dynamically allocated by the BBI-SS PAS. Each trace request is assigned its own trace log data sets.
trace log directory (TDIR)

A VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information.

transaction

A specific set of input data that initiates a predefined process or job. A transaction can consist of one or more components.

Transaction Accountant

An MainView for IMS Offline component that produces cost-accounting and user charge-back records and reports.

transaction correlation

Transaction correlation is the process of identifying all the components that belong to a particular complex transaction starting with any component of that transaction.

MainView Transaction Analyzer can correlate transactions automatically, even those that span different systems and multiple transactional environments.

You can display correlated components in a MainView table view or as a graphical flow chart in MainView Explorer.

transactional environment

A transactional environment is any application that processes transactional units of work. MainView Transaction Analyzer identifies units of work running on

- CICS Transaction Gateway
- CICS Transaction Server
- DB2
- IMS
- WebSphere MQ
TS

See terminal session (TS) on page 381.

TSO workload

A group of units of work consisting of address spaces running TSO sessions.

U

UAS

See user address space (UAS) on page 384.

UBBPARM

See parameter library on page 372.

UBBPROC

See procedure library on page 374.

UBBSAMP

See sample library on page 376.

unit of recovery (UOR)

A UOR can be viewed as the set of changes to recoverable resources made within a specific process or task running in a single execution environment.

unit of work (UOW)

A UOW is a collection of one or more UORs that are related by the fact that the resources for each are committed or rolled back as an atomic unit. A UOW begins when a transaction starts or after a user-requested syncpoint. It ends either at a user-requested syncpoint or at the end of a transaction.

In MainView Transaction Analyzer, UOWs are referred to as components.
You can consider unit of work as a synonym for ACID transaction on page 345.

user address space (UAS)

An address space that runs a MainView terminal session in TSO, VTAM, or EXCP mode.

user BBPROF

See profile library on page 374.

view

The formatted data within a MainView window, acquired from a product as a result of a command or action. A view consists of two parts: query and form. See also form on page 358, job activity view on page 361, query on page 375.

view command

The name of a view that you type on the COMMAND line to display that view.

view command stack

An internal collection of up to 10 queries. For each command, the stack contains the filter parameters, sort order, context, product, and time frame that accompany the view.

view definition

The meaning of data that appears online, including source of data, selection criteria for data-field inclusion and placement, data format, summarization, context, product, view name, hyperlink fields, and threshold conditions.

view Help

Embedded information describing the purpose of a specific view. To display view Help, place the cursor on the view name on the window information line and press PF1 (HELP).
window

An area of the MainView screen in which views and resources are presented. A window has visible boundaries and can be smaller than or equal in size to the MainView window area. See also active window on page 345, alternate window on page 346, current window on page 353, MainView window area on page 368.

window number

A sequential identifier assigned by MainView to each window when it is opened. The window number is the second character in the window status field. See also window status field on page 385.

window information line

The top border of a window. It shows the window identifier, the name of the view displayed in the window, the system, the scope, the product reflected by the window, and the time frame for which the data in the window is relevant. See also window status field on page 385.

window status

A one-character letter that indicates when a window is ready to receive commands, is busy processing commands, is not to be updated, or contains no data. The window status also indicates when an error has occurred in a window. The window status is the first character in the window status field. See also window information line on page 385, window status field on page 385.

window status field

An area on the window information line that shows the current status and assigned number of the window. See also window number on page 385, window status on page 385.

windows mode

A collection of one or more MainView product views on a screen that can be divided into a maximum of 20 windows. A window information line defines the top border of each window. See also full-screen mode on page 358.
WLM workload

In goal mode in MVS/SP 5.1 and later, a composite of service classes. MainView for z/OS creates a workload for each WLM workload that is defined in the active service policy.

workflow

A measure of system activity that indicates how efficiently system resources are serving the jobs in a workload.

workload

(1) A systematic grouping of units of work (for example, address spaces, CICS transactions, IMS transactions) according to classification criteria established by a system administrator. (2) In z/OS, a group of service classes within a service definition.

workload definition

A group of units of work that is created through the WKLIST view. A definition contains a unique name, a description, an initial status, a current status, and selection criteria by which address spaces are selected for inclusion in the workload. See also Workload Definition Facility on page 386.

Workload Definition Facility

In MainView for z/OS, the WKLIST view and its associated dialogs through which workloads are defined and service objectives are set.

workload activity view

Data that shows workload activity as the workload accesses system resources. A workload activity view measures workload activity in terms of resource consumption and how well the workload activity meets its service objectives.

Workload Analyzer

The online data-collection and display services that are used to examine IMS workloads and determine problem causes.

workload delay view
Data that shows workload performance as the workload accesses system resources. A workload delay view measures any delay a workload experiences as it contends for resources.

**Workload Monitor**

The online data-collection services that are used to monitor IMS workloads and issue warnings when defined thresholds are exceeded.

**workload objectives**

The performance goals for a group of units of work defined in WKLIST. Objectives can include measures of performance such as response times and batch turnaround times.

**Z**

**z/OS PAS**

Formerly MVS PAS and OS/390 PAS. See product address space (PAS) on page 374.

**zAAP**

See zSeries Application Assist Processor (zAAP) on page 387.

**zFS**

See zSeries file system (zFS) on page 388.

**zFS aggregate**

A single data set that contains multiple file systems. A zFS aggregate can contain one, two, or more zFSs. A zFS aggregate is a Virtual Storage Access Method Linear Data Set (VSAM LDS). After the zFS aggregate is defined and formatted, one or more zFSs can be added to the aggregate. See also hierarchical file system (HFS) on page 358.

**zSeries Application Assist Processor (zAAP)**

A specialized processor from IBM that executes Java applications to reduce the workload on general purpose Central Processors (CPs).
zSeries file system (zFS)

A UNIX file system that contains files and directories that can be accessed with APIs. A zFS can be mounted into the z/OS UNIX hierarchy in addition to other local and remote file system types, such as HFS and TFS (temporary file system).
Index

_ALL context 185
_CURRSYS context 185
@BBXINIT member 303
/ character for operator commands 145
#DEFHFS member 241
#NORMAL member 243
#UIM member 237

A

AAOGMExx member 77
abend diagnostics, BBI 295
ACID transaction
  definition 345
action
  definition 345
active window
  definition 345
ADD command
  CAS definitions 102
  LOGGERxx members 131
  MainView Logger system entries 134
  SSI context definitions 196
address spaces, managing
  BBI-SS PAS 57
  CAS 21
  z/OS PAS 37
administrative view
  definition 345
AF_UNIX 305
AFF_TIMEOUT_SECS parameter 236, 243
aggregate
  definition 346
ALL context 183, 185
ALLOW_NETCMD 253
ALLOW_NETCMD parameter 236, 243
ALLOW_SAF_RC4 element 322, 329
ALT WIN field

B

batch workload
  definition 347
BBHDEF data set 273
BBI
  architecture 14
  control commands 77
  definition 347
diagnostic facilities 291
LOGREC records 292
BBI-formatted dump 293
BBI-SS PAS
  cold start 59
cross-system communication 116
definition 347
DISPLAY command 77
displaying status 62
dump data set 60
free start 59
region size 57
stall interval parameter 61
starting before JES 60
starting timer-driven requests 62
warm start 59
BBIILOG started task 69
BBIIMAGL data set 71
BBIIMAGn data sets 69
BBISP00 member 62
BJJNT00 member 150, 172
BJJRNLL data set 68
BJJRNLn data sets 64
BBISPRNT data set 68
BBISSP00 dump parameters 294
BBISSP00 member 59, 61, 64
BBITSF00 dump parameters 294
BBIUSR01 user exit 64
BBIUSR02 user exit 69
BBIXSF00 dump parameters 294
BBKEYS member 77
BBMCAS startup procedure 22
BBMSLPBM member 343
BBMTXP00 member 172
BBPARM
definition 347
  LOGGERxx members 129
  MVINGlxx members 129
BBPARM data set
  AAOGMExx member 77
  BBJNT00 member 150, 172
  BBISP00 member 59, 64
  BBKEYS member 77
  BBMTXP00 member 172
  BBTTCPnn member 77
  CMFCPM04 member 47
  CMFIPM01 member 47
  PWSCPM01 member 47
BBPROC
definition 347
BBPROF
definition 347
BBPROF data set, BBITSP00 member 294

BBSAMP
definition 348
BBSAMP data set
  @BBXINIT member 303
  BBMSLPBM member 343
  DLOGJCL member 64
  IMFMSG member 298
  LOGRECS member 292
  MVLXPJCL member 146
  PLOGJCL member 64
  SMLXIT member 64
  SSJCL member 57
  BBTTCPnn member 77
BBV
definition 348
BBXS
definition 348
BBXS subsystem
  CSA usage 302
  initializing 303
  installing 302
  overview 301
  running multiple products 302
BMC Subsystem Services, See BBXS subsystem 301
BMC UIM Server Commands web page 253
border
definition 348
bottleneck analysis
definition 348
BPXPRMxx 305

C

CA-Disk
definition 348
CANCEL command
  CAS definitions 106
  MainView AutoOPERATOR 77
  MainView Logger system entries 137
  SSI context definitions 198
  target context definitions 170
  z/OS PAS 54
CAPS parameter, CAS 22
CAS
  definition 348
  CAS definitions
adding 102
canceling changes 106
changing 104
deploying 109
editing 102
managing 99
recovering 107
saving changes 109, 174
CAS, See coordinating address space (CAS) 21
CASACT view 111
CASDEF view 100
CASINFO view 112
CASPERM registry
  changing data set name 32
  creating 29
  disabling 31
  enabling 31
  overview 29
CASPERMS view 29
CFMON
  definition 348
CHAnge command
  MainView Logger system entries 134
  target context definitions 168
chart
  definition 348
CICSplex
  definition 349
CMF MONITOR
  definition 349
CMF MONITOR Analyzer
  definition 349
CMF MONITOR Extractor
  CPM mode samplers 39
  definition 349
  initializing 39
  multiple z/OS PASs 52
  overriding the default member 48, 50
CMF MONITOR Online
  definition 349
CMF Type 79 API
  definition 349
CMFCPM04 member 47
CMFIPM01 member 47
CMFIPMnn members
  contents 50
  creating 50
CMFMON
  definition 349
CMFXDS00 member 52
CMRCTG parameter 77
CMRDETL
  definition 350
CMRPRBT parameter 77
CMRSECU parameter 77
CMRSOPT parameter 77
CMRSTATS
  definition 350
COLD parameter, CAS 22
cold start
  BBI-SS PAS 59
  coordinating address space 22
collection interval
  definition 350
column
  definition 350
command delimiter
  definition 350
COMMAND element 329
COMMAND line
  definition 350
Command MQ Automation D-S
  definition 351
Command MQ Automation S-390
  definition 351
Command MQ for D/S
  definition 351
Command MQ for S-390
  definition 350
commands
  BBI control 77
  DBC 317
  DIAG 317
  operator 145
  RESETPRINT 317
  RESETTRACE 317
  STOP 317
  TRACEFLUSH 317
  TRACEOFF 317
  TRACEON 317
common registry, MainView Infrastructure 231
common service area (CSA), BBXS subsystem 301
COMMON STORAGE MONITOR
  definition 351
COMMORDR parameter, CAS 22
communication
  BBI-SS PAS 116
  CAS-to-CAS 98
cross-system 97
compatibility mode aggregate
definition 351
complex transaction
definition 351
component
definition 352
COMPONENT element 328
composite workload
definition 352
Comprehensive Management Facility MONITOR (CMF MONITOR)
definition 352
CONACT view
accessing 205
overview 202
CONACTD view
accessing 206
overview 202
CONACTZ view
accessing 203
overview 202
CONDEFF view
accessing 193
overview 186
CONDEFD view
accessing 194
overview 186
CONDEFFL view
accessing 191
overview 186
configuring UIM server 243
constituent workload
definition 352
contention
definition 353
context
definition 353
CONTEXT command
definition 353
context definitions
deploying 174
securing 181, 208
single system image (SSI) 183
target 151, 159
CONTEXT element 327
contexts
single system image (SSI) 183
target 149
CONTEXTS element 327
control commands, BBI 77
control statement
definition 354
conventions, documentation 12
CONVXCF parameter, CAS 22
coordinating address space (CAS)
applying maintenance 32, 34
CASPERM registry 29
communication between 98
defining SSID 22
definition 353
displaying information 112
initializing in multisystem environment 27
managing definitions 99
overview 15, 21
restarting 34
START command parameters 22
starting 21
stopping 32
TCP/IP communication 105, 338
termination message 32
UIM server connection 239, 251
COPY command
SSI context definitions 195
target context definitions 169
coupling facility monitoring (CFMON)
definition 353
CPM monitor 39
CPMCMFnn members
contents 48
creating 48
creating UIM server 235
cross-system communication
BBI-SS PAS 116
CAS-to-CAS 98
managing CAS 99
overview 97
quiescing CAS 114
starting BBI-SS PAS 116
starting CAS 113
stopping CAS 115
with historical data sets] 210
Cross-System Coupling Facility (XCF)
COMMORDR parameter 22
CONVXCF parameter 22
DFLTGRP parameter 22
SPCF parameter 22
cross-system data server (XDS) 39
CSA (common service area), BBXS subsystem 301
current data
definition 354
current window
definition 353
CURRSYS context 185

D

D RULES command, BBI control 77
DASD
  definition 354
data collector
definition 354
Data Facility Storage Management System (DFSMS)
definition 354
Data Set Optimizer (DSO)
definition 354
DATAACLAS element 315
DBC
  stopping 316
DBC commands
  DIAG 317
  DISPAGENTS 317
  DISPSNS 317
  RESETPRINT 317
  RSETTRACE 317
  STOP 317
  TRACEFLUSH 317
  TRACEOFF 317
  TRACEON 317
DBC control point resource profile 324
DBC security 320
DBC security parameters 325
DBC started task
  authorizing 322
  DBCPARMS 305
  DBCPRINT 307
  DBCSECUR 307
  DOMPARMS 305
  DOMMTPLT 307
  JCL 305
  starting 305
  STEPLIB 307
  SYSPRINT 307
DBC subsystem
  starting 305
  stopping 305
  working with 305
DBCPARMS DD statement 305, 321
DBCPARMS parameters 305, 307
DBCPRINT DD statement 307
DBCSECUR DD statement 307, 321, 325
DBCSECUR element 326
defined SSI contexts 185
definition tags, online Help 279
DELeTe command
  CAS definitions 107
  LOGGERxx members 132
  SSI context definitions 199
target context definitions 170
delta mode
definition 354
deploying
  CAS definitions 109
  single system image (SSI) contexts 201
target contexts 174
DFLTGRP parameter, CAS 22
DFSMS
definition 355
DIAG command 317
DIAGMSG view 261
diagnostic facilities, BBI 291, 295
diagnostic message modes
  GEMM 262
  GESTR 262
  GSDM 262
  GSSM 262
  GSSTR 262
  LEMM 262
  LESTR 262
  LSEMM 262
  LXDM 262
  SAFTRACE 262
  SAFTRC 262
  WSXASTR 262
diagnostic messages
  BBI 292
  Plex Manager 261
DIAGSESS view
  halting VTAM communication 115
  quiescing VTAM communication 114
  stopping an z/OS PAS 54
DIDOCs console 96
direct access storage device (DASD)
definition 355
display
definition 355
DISPLAY command
  BBI control 77
  VTAM control 116
DLOGJCL member
journal record format 297
printing journal log 64
printing terminal session log 68

DMR
definition 355
DMRBEX00 parameter 77

DMS
definition 355
DMS2HSM
definition 355
documentation information 11
DOMPARMS DD statement 305
DOMTMPLT DD statement 307
DPRREPOS element 313

DSLIST view
column information 216
historical database 215
line commands 218
valid commands 215

DSO
definition 355

DUMP
BBI control command keyword 77
CAS Start parameter 22
dump data set, BBI-SS PAS
opening 60
resetting 77

DUMPS keyword, SET command 77
dumps, BBI diagnostic 293, 294
dynamic SSI contexts 185
dynamic targets 150, 172

E

EasyHSM
definition 355
EasyPOOL
definition 355
EasySMS
definition 356

EDIT command
CAS definitions 102
LOGGERxx members 131
MainView Logger system entries 134
SSI context definitions 195
element
definition 356
element Help
definition 356

EMM parameter, CAS 22
ENCRYPTION_LEVEL parameter 236, 243
Enhanced Journal Facility 64

Enterprise Storage Automation
definition 356
error messages
BBI diagnostic 292
BBI-SS PAS SELECT statement 72
journal log 64
event
definition 356
Event Collector
definition 356
EXEC keyword
START command 77
STOP command 77

EXEC, MainView AutoOPERATOR
canceling delayed actions 77
cancelling 77
displaying Automation Logger status 77
displaying delayed actions 77
displaying statistics 77
starting 77
stopping 77
expand
definition 356
exporting MainView Logger records 146
extended diagnostic mode (XDM)
CAS startup procedure 22
displaying messages 140
z/OS PAS startup procedure 39
extended message mode (EMM) 22
extractor
definition 356
extractor interval
definition 357

F

fast path
definition 357
field
definition 357
field Help
definition 357
filter
definition 357
fire
definition 357
fixed field
definition 357
flow chart
definition 358
FMID element 316
form
definition 358
formatted dump, BBI 293
free start, BBI-SS PAS 59
full-screen mode
definition 358

G
GEMM message mode 262
GENERATE command 166
GESTR message mode 262
GET command 77
global command
definition 358
global options, MainView Explorer 120
GME keyword
DISPLAY command 77
RESET command 77
START command 77
STOP command 77
graph
definition 358
graphical user interface (GUI) 17
GROUP element 311
GSDM message mode 262
GSSM message mode 262
GSSTR message mode 262

H
HALt command, CAS 115
HFS
definition 358
HFS file system
MainView Infrastructure common registry 231
UIM server 241
HFS_DATASET parameter 236, 243
hierarchical file system (HFS)
definition 358
hierarchical storage management (HSM)
definition 359
hilevel
definition 359
historical data
definition 359
historical data set
adding dynamically 225
allocating 213
archiving 228
closing 223
cross-system communication 210
data recording frequency 211
deallocating 227
definition 359
displaying 216
emptying 224
managing 209
number required 213
read/write access 220, 222
recording order 218
recording status 216
starting recording 211
historical database 210
definition 359
HLQ element 326
host server, MainView 119
HSM
definition 359
hyperlink
definition 359
hyperlinks, online Help 282, 283

I
ICHERCDE macro 323
ICHRFRTB macro 324
IDCAMS utility 213, 228
IGNORE element 315
ILOGJCL member
image log record format 298
printing image log 72
IMAGE keyword
START command 77
STOP command 77
SWITCH command 77
image log
BBI-SS PAS 69
BBIIMAGn data sets 69
definition 360
DISPLAY command 77
maintenance 69
printing 72
record format 298
terminal session 71
IMFBEX00 parameter 77
IMFMSG member 298
IMRPRINT program 72
IMS Resource Utilization File (IRUF)
definition 360
IMSPlex System Manager (IPSM)
definition 360
initializing
BBXS subsystem 303
CAS in multisystem environment 27
CMF MONITOR Extractor 39
z/OS PAS data collectors 39
INSTALL command
CAS definitions 108
SSI context definitions 200
target context definitions 172
target definitions 155
interval data
definition 360
InTune
definition 361
IPM monitor 39, 50
IRUF
definition 361

J

JES filter, DISPLAY command 77
JES queue information, displaying 269
job activity view
definition 361
job status information, displaying 269
JOBLSTO view 269
JOBSTAT view 269
journal
definition 361
JOURNAL keyword
START command 77
STOP command 77
SWITCH command 77
journal log

K

Katakana terminal support 22
KEYS keyword 77

L

LEMM message mode 262
LESTR message mode 262
line command
definition 361
line command column
definition 362
LINK keyword
DISPLAY command 77
START command 77
STOP command 77
VTAM control 116
link tags, online Help 283
links between BBI-SS PASs
controlling 116
failed connection 57
LIST file 72
list tags, online Help
definition 279
ordered 279
parameter 283
simple 279
summary 283
unordered 279
LOCATE command, MainView AutoOPERATOR 77
LOG DISPLAY 77, 96
Log Edit
definition 362
log files, MainView Logger
displaying 139
overview 139
Log Message Detail dialog 140
log messages, MainView Logger
displaying 140
locating by criteria 143
locating by time 144
logger,. See MainView Logger
LOGGERxx members
adding 131
deleting 132
displaying 130
editing 131
overview 128
saving 138
logging 127
BBI control commands and responses 96
BBI-SS PAS 64
BBI-SS PAS image 69
overview 63
terminal session image 71
terminal session journal 68
LOGJRNL view 140
LOGLIST view 139
LOGMSG view 140
LOGPROF command 143
LOGRDEF view 132
LOGRDEFD view 132
LOGRDEFL view 130
LOGREC records, BBI 292
LOGRECS member 292
LOGRESP view 145
LOGRPT view 140
logspace
definition 362
logspace parameters 128
LOGTIME command 144
LSEMM message mode 262
LSSTR message mode 262
LXDM message mode 262

M
maintenance, applying
CAS 32
MainView
definition 362
MainView Alarm Management
definition 364
MainView Alternate Access 17
definition 364
MainView Application Program Interface (MVAPI)
definition 362
MainView AutoOPERATOR
canceling delayed actions 77
cancelling an EXEC 77
definition 362
displaying AOLOG 77
displaying delayed actions 77
displaying EXEC statistics 77
displaying TapeShare partners 77
GET command 77
RESET command 77
starting before JES 60
MainView control area
definition 364
MainView display area
definition 365
MainView Explorer
definition 363
overview 17
setting global options 120
starting the host server 119
stopping the host server 120
MainView Explorer flow chart
definition 365
MainView for CICS
definition 365
MainView for DB2
definition 364
MainView for DBCTL
definition 365
MainView for IMS Offline
definition 364
MainView for IMS Online
definition 364
MainView for IP
MainView for Linux - Servers
  definition 364
MainView for MQSeries
  definition 365
MainView for OS/390
  definition 365
MainView for UNIX System Services
  definition 366
MainView for VTAM
  definition 363
MainView for WebSphere Application Server
  (formerly MainView for WebSphere )
  definition 363
MainView for WebSphere MQ (formerly MainView for MQSeries)
  definition 363
MainView for WebSphere MQ Integrator
  definition 363
MainView for z/OS
  definition 365
MainView Infrastructure
  common registry 231
  overview 13
  UIM server connection 239, 250
MainView Logger
  controlling operation 146
  definition 363
  definition members defined 128
  exporting log records 146
  issuing operator commands 145
  log files 139
  Log Message Detail dialog 140
  log messages 140
  logger definition member entries 132
  logger definition members 130
  LOGGERxx members 130
  LOGJRNRL view 140
  LOGLIST view 139
  LOGMSG view 140
  LOGPROF command 143
  LOGRDEF view 132
  LOGREDFL view 130
  LOGRESP view 145
  LOGRPT view 140
  logspaces defined 128
  LOGTIME command 144
  MODIFY commands 146
  NGL commands 146
  overview 127
  started task procedures 129
MainView Selection Menu
  definition 366
MainView SRM
  definition 366
MainView SRM Allocation
  definition 366
MainView SRM Automation
  definition 366
MainView SRM DMS2HSM
  definition 367
MainView SRM EasyHSM
  definition 366
MainView SRM EasyPOOL
  definition 367
MainView SRM EasySMS
  definition 367
MainView SRM Enterprise Storage Automation
  definition 367
MainView SRM Reporting
  definition 367
MainView SRM SG-Auto
  definition 367
MainView SRM SG-Control
  definition 367
MainView SRM StopX37/II
definition 367
MainView SRM StorageGUARD
definition 367
MainView Storage Resource Manager (SRM)
definition 368
MainView SYSPROG Services
definition 368
MainView SYSPROG services, invoking 52
MainView VistaPoint
definition 368
MainView window area
definition 368
MAXSOCKETS 305
messages
   BBI-SS PAS SELECT statement 72
   Plex Manager diagnostic 261
   restricting journal log 64
MGMTCLAS element 315
MODE element 312
MODIFY (F) command
   BBI control commands 94
   MainView Logger operation 146
   z/OS PAS operation 52
monitor
definition 368
MTALOG
definition 368
Multi-Level Automation (MLA)
definition 368
MVALARM
definition 369
MVAPIS
definition 369
MVCICS
definition 369
MVDB2
definition 369
MVDBC
definition 369
MVIMS
definition 369
MVIP
definition 369
MVLNX
definition 369
MVLOG keyword
   DISPLAY command 77
   RESET command 77
   START command 77
   STOP command 77
   MVLXPJCL member 146
MVMQS
definition 369
MVMVS
definition 369
MVS PAS
definition 370
MVS PAS.. See z/OS PAS
MVScope
definition 370
MVSRM
definition 370
MVSRMHSM
definition 370
MVSRMSGC
definition 370
MVSRMSGD
definition 370
MVSRMSGP
definition 370
MVUSS
definition 370
MVVP
definition 370
MVVTAM
definition 371
MVWEB
definition 371

N
NAME element 313
nested Help
definition 371

O
object
definition 371
OMVS initialization 305
OMVS segment 323
OMVS segments for TCP/IP 105
OMVS workload
definition 371
online Help
BBI control HELP command 77
creating your own 273
customized Help data set 273
definition 371
search order for displaying 275
tag language 275, 283
operator commands, issuing 145
OPTIONS element 310
originating subsystem
definition 371
originating system
definition 372
OS/390 and z/OS Installer
definition 372
OS/390 PAS
definition 372

P

parameter library
definition 372
parameter tags, online Help 283
PARAM keyword 77
PARTNERS keyword 77
PAS
definition 372
PAS. See product address space (PAS)
password keys, displaying 77
performance group (PRGP) workload
definition 373
PERFORMANCE MANAGER
definition 372
Performance Reporter
definition 373
PIID element 316
Plex Manager 15
definition 373
diagnostic messages 261
displaying SSI contexts 202
displaying targets and products 176
navigating 180
status views 178
PLEX view 178
PLEXAREA view 178
PLEXOVER view 178
PLOGJCL member
journal record format 297
printing journal log 64
pop-up display

query
definition 375
QUIEsce command, CAS 114
quiescing VTAM communication 114

R

RACF resource class 323
RDEFINE FACILITY commands 324
real-time data
definition 375
record format
image log 298
journal log 297
recording status, historical data set
changing order 212
displaying 216
region size, BBI-SS PAS 57
registries
   CASPERM 29
      MainView Infrastructure common 231
related publications 11
RESET command 77
RESETPRINT command 317
RESETTRACE command 317
Resource Analyzer
definition 375
Resource Monitor
definition 375
RESOURCE_CLASS element 328
RESOURCE_NAME element 326
retrieving historical data 228
RMF considerations 39
row
definition 375
RTCS
definition 375
RULE keyword, SET command 77
RULES keyword
   DISPLAY command 77
   RESET command 77
   START command 77
   STOP command 77
RULESET keyword 77
Runtime Component System (RTCS) 17
definition 375
RxD2
definition 376

S
SAFTRACE message mode 262
SAFTRC message mode 262
sample cycle
definition 376
sample library
definition 376
sampler
definition 376
SAVE command
   SSI context definitions 200
target context definitions 173
SBBPROF
definition 376
scope
definition 377
screen definition
definition 377
screen image record 68
screen printing facility 68
security 320
SELECT statement 72
selection view
definition 377
service class workload
definition 377
service objective
definition 377
service point
definition 377
service request block (SRB)
definition 378
service select code
definition 378
session
definition 378
SET command 77
SETROPTS command 324
SG-Auto
definition 378
SG-Control
definition 378
single system image (SSI)
definition 378
single system image (SSI) contexts
   adding 196
      cancelling changes 198
      changing 197
      deploying 201
      displaying 193
      displaying status 202
      managing 190
      overview 183
      planning 185
      securing 208
      types 185
Skeleton Tailoring Facility
definition 378
SLOGJCL member 68
SMF ID 39
SMFID element 327
SMLXIT member 64
SMLXIT user exit 64
SNAP dump, BBI 293
SOF keyword, DISPLAY command 77
SPCF parameter, CAS 22
SRB
  definition 379
SSI
  definition 379
SSI context
  definition 379
SSI contexts,. See single system image (SSI) contexts
SSID element 310
SSID parameter
  coordinating address space 22
  z/OS PAS 39
SSIJCL member 57
stall interval parameter, BBI-SS PAS 61
START command
  BBI control 77
  BBI-SS PAS 57
  BBI-SS PAS communication 116
  image logging 77
  z/OS PAS 39
Started Task workload
  definition 379
starting
  BBI-SS PAS 57
  MainView host server 119
  UIM server 252
  VTAM communication 113, 116
  z/OS PAS 39
statistics interval
  definition 379
status, displaying
  SSI contexts 202
  TapeShare partners 77
  targets and systems 176
stem variables
  definition 379
STEPLIB DD statement 307
STOP command 317
  BBI control 77
  coordinating address space 32
  image logging 77
  VTAM links between BBI-SS PASs 116
  z/OS PAS 54
stopping
  BBI-SS PAS 62
  coordinating address space 32
  MainView AutoOPERATOR EXECs 77
  MainView host server 120
  UIM server 252
  VTAM links between BBI-SS PASs 116
  z/OS PAS 54
stopping DBC 316
StopX37-II
  definition 379
StorageGUARD
  definition 379
STORCLAS element 314
SUB parameter, CAS 22
SUBSYS element 329
subsystem command recognition character 94
subsystem ID
  coordinating address space 22
  z/OS PAS 39
SUBSYSTYPE element 313
summary view
  definition 380
SVC dump, BBI 293
SWITCH command 77
SYSABEND data set 293
SYSLOG messages, displaying 140
SYSNAME context 185
SYSPLEX context 185
SYSPRINT DD statement 307
SYSPROG services
  definition 380
System Authorization Facility (SAF) interface 321
SYSTEM context 185
system entries, MainView Logger
  adding 134
  cancelling edits 137
  changing 134
  deleting 136
  displaying 132
  displaying deleted 137
  editing 134
  overview 128
system resource
  definition 380
SYSUDUMP data set 293
T
tag language, online Help
cautions and warnings 283
commonly used tags 278
control words 277, 283
definition lists 283
definition lists, 283
example 276
figures 283
headings 283
highlighting 283
imbedding Help text 283
introduction 275
lists 279
margins 283
notes 283
paragraphs 283
rules 277
tape devices
  checking path validity 77
  varying status 77
TapeShare partners 77
target
definition 380
target context
definition 380
target context definitions
  adding definitions 167
  adding members 163
  cancelling changes 170
  defining 154
  deleting members 162
  deploying 174
  displaying definitions 162, 164
  displaying members 160, 161
  editing 165
  installing 172
  managing 159
  planning 152
  securing 181
target contexts
  defining 151
  displaying status 176
  dynamic 150, 172
  overview 149
  specifying 150
TARGET parameter in BBITSP00 150
TASCOOSTR
definition 380
task control block (TCB)
definition 380
TCB
definition 381

TCP/IP communication
  CAS startup restrictions 105
  COMMORDR parameter 22
  defining 105, 338
  OMVS segments 105
  secure ports 105
  stopping the CAS to activate 32, 338
TDIR
definition 381
terminal session
  image log 71, 72
  journal log 68
  overview 17
terminal session (TS)
definition 381
terminating
  coordinating address space 32
  MainView AutoOPERATOR EXECs 77
  MainView host server 120
  UIM server 252
  VTAM links between BBI-SS PASs 116
  z/OS PAS 54
TGTDEF view
  accessing 155, 162
  overview 154
TGTDEFD view
  overview 154
TGTDEFL view
  accessing 160
  overview 154
threshold
definition 381
TLDS
definition 381
TO_VALUE element 327
total mode
definition 381
trace
definition 381
trace log data set (TLDS)
definition 381
trace log directory (TDIR)
definition 382
TRACE member 249
TRACEFLUSH command 318
TRACEOFF command 317
TRACEON command 318
transaction
definition 382
Transaction Accountant
transaction correlation definition 382
transactional environment definition 382
troubleshooting, UIM server 258
TS definition 383
TS. See terminal session
TSLOG, batch allocation to BBISPRNT 68
TSO workload definition 383

U
U4095 abends, suppressing 343
UAS definition 383
UAS. See user address space (UAS) 17
UBBPARM definition 383
UBBPROC definition 383
UBBSAMP definition 383
UIM server

UIM server
#DEFHFS member 241
#NORMAL member 243
#UIM member 237
accessing BMC UIM Server Commands web page 253
affinity timeout 236, 246
CAS connection 239, 251
configuring 243
creating 235
encryption of user credentials 236, 247
HFS data set 236, 241, 246
HFS root directory 236, 248
MainView Infrastructure connection 239, 250
network browser command interface 236, 247
operating 250
overall tracing option 249, 256
overview 235
port number 236, 245
refreshing content 257
security authorization timeout 236, 245, 255
specific tracing options 250, 256
started task procedure 237
starting 252
startup configuration member 236
stopping 252
TRACE member 249
troubleshooting 258
UIM$PCH member 239
verifying operation 253
viewing active users 255
UIM$PCH member 239
UNDelete command
target context definitions 171
unit of recovery (UOR) definition 383
unit of work (UOW) definition 383
user address space (UAS) 17 definition 384
user BBPROF definition 384
user exits
BBIUSR01 64
BBIUSR02 69
SMLXIT 64
user IDs
authorizing 323
USS_HFS_ROOT parameter 236, 243
V

view
definition 384
view command
definition 384
view command stack
definition 384
view definition
definition 384
view Help
definition 384
VOLUMES element 314
VTAM communication
  COMMORDR parameter 22
  quiescing 114
  subsystem connection failure 57

W

warm start, BBI-SS PAS 59
web browser 17
window
definition 385
window information line
definition 385
window number
definition 385
window status
definition 385
window status field
definition 385
windows mode
definition 385
WLM element 312
WLM workload
definition 386
workflow
definition 386
workload
definition 386
workload activity view
definition 386
Workload Analyzer
definition 386
workload definition
definition 386
Workload Definition Facility
definition 386
workload delay view
definition 386
Workload Monitor
definition 386
workload objectives
definition 387
WSXASTR message mode 262
WTO messages
  BBI control commands 96
  BBI diagnostic 292, 295
  WTOUPPERCASE element 313

X

XCF (Cross-System Coupling Facility)
  COMMORDR parameter 22
  CONVXCF parameter 22
  DFLTGRP parameter 22
  SPCF parameter 22
  XCFGROUP element 312
XDM (extended diagnostic mode)
  CAS startup procedure 22
  displaying messages 140
  z/OS PAS startup procedure 39
XDS (cross-system data server) 39
XDS parameter, z/OS PAS 39

Z

z/OS PAS
  definition 387
z/OS product address space (PAS)
  overview 37
  production and test 52
  relationship to CAS 37
  running concurrently 52
  started task parameters 39
  starting 39
  stopping 54
zAAP
  definition 387
zFS. See zSeries file system
zFS aggregate
  definition 387
zFS file system 231
zSeries Application Assist Processor (zAAP)
  definition 387
zSeries file system 387, 388