

MAINVIEW[®] Common Customization Guide

Version 4.1

July 15, 2002



Copyright 2002 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. All other registered trademarks or trademarks belong to their respective companies.

THE USE AND CONTENTS OF THIS DOCUMENTATION ARE GOVERNED BY THE SOFTWARE LICENSE AGREEMENT ENCLOSED AT THE BACK OF THIS DOCUMENTATION.

Restricted Rights Legend

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b) or DFARS 227.7202, 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.

Contacting BMC Software

You can access the BMC Software Web site at <http://www.bmc.com>. From this Web site, 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

Telephone 713 918 8800 or
800 841 2031

Fax 713 918 8000

Outside United States and Canada

Telephone (01) 713 918 8800

Fax (01) 713 918 8000

Customer Support

You can obtain technical support by using the Support page on the BMC Software Web site or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software.”

Support Web Site

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at <http://www.bmc.com/support.html>. From this Web site, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software 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 800 537 1813. Outside the United States and Canada, please contact your local support center for assistance. To find telephone and e-mail contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at www.bmc.com/support.html.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem 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, such as `file system full`
 - messages from related software

Contents

About This Book	xiii
Chapter 1	Customizing MAINVIEW
	Understanding the MAINVIEW Infrastructure 1-1
	Coordinating Address Space (CAS)..... 1-3
	Product Address Space (PAS) 1-3
	User Address Space (UAS) 1-5
	Understanding MAINVIEW Product Customization..... 1-6
Chapter 2	Migration Considerations
	Migrating CAS Definitions..... 2-2
	Running Mixed Versions of MAINVIEW Infrastructure..... 2-2
	On Different MVS Systems 2-3
	On the Same MVS System..... 2-4
Chapter 3	Using MAINVIEW Product Libraries
	Parameter Libraries 3-6
	BBPARAM 3-6
	UBBPARAM..... 3-6
	Customized IMS Parameters 3-7
	Sample Libraries 3-8
	BBSAMP 3-8
	UBBSAMP..... 3-9
	Profile Libraries 3-9
	BBPROF..... 3-10
	SBBPROF 3-10
	User BBPROF 3-11
	Procedure Libraries (MAINVIEW AutoOPERATOR Only) 3-11
	BBPROC 3-11
	UBBPROC 3-12
	View, Screen, and View Help Distribution Libraries 3-13
	Site Libraries 3-13
	User Libraries 3-14
	Shared Libraries 3-15

Library Concatenation	3-16
Link Library	3-16
Product Support Data Sets.....	3-16
Image and Journal Logs	3-18
Product Library Customization Summary	3-19

Chapter 4 Preparing for Customization

Chapter 5 Customizing the MAINVIEW Environment

Step 1: (Required) Specify Jobcards and Other Operational Defaults ..	5-2
Step 2: (Optional) Review Virtual Storage Requirements	5-3
Step 3: (Optional) Generate JCL to Run Installation Verification Program	5-4
Step 4: (Optional) Implement GDDM/PGF Support	5-6
Step 5: (Optional) Implement Katakana Terminal Support	5-7
Step 6: (Required) Apply Passwords with the Product Authorization Utility	5-9
Step 7: (Required) Create Site Data Sets for Use with MAINVIEW Products	5-10
Step 8: (Required for Security) Create Site Security Data Set	5-13
Step 9: (Required) APF-Authorizing the BMC Software Load Library ..	5-14
Step 10: (Optional) Add the BMC Software Load Library to your System	
Link List	5-16
Step 11: (Required) Create CLIST for Invoking MAINVIEW Products	5-18
Step 12: (Optional) Add TSF Authorization for Products to Run Under ISPF	5-20
Step 13: (Optional) Reload All BBX Services	5-21
Step 14: (Required) Create the CAS (Coordinating Address Space) Startup	
Procedure	5-23
Step 15: (Optional) Create MAINVIEW Explorer Host Server Startup	
Procedure	5-25
Step 16: (Optional) Migrate CAS Definitions	5-27
Step 17: (Required) Specify MAINVIEW Alarm Manager Initialization	
Parameters	5-28
Step 18: (Required) Allocate MAINVIEW Alarm Manager Journal Logs	5-30
Step 19: (Required) Allocate MAINVIEW Alarm Manager BBIDIV Data Set	
.....	5-32
Step 20: (Required) Create the MAINVIEW Alarm Manager PAS Startup	
Procedure	5-33
Step 21: (Optional) Create MVS PAS Historical Data Sets	5-35
Step 22: (Required) Create MVS PAS PARMFILE Data Set	5-38
Step 23: (Required) Create the MVS PAS Startup Procedure.....	5-39
Step 24: (Optional) Create BBI-SS PAS Historical Data Sets	5-41
Step 25: (Optional) Create BBI-SS PAS PARMFILE Data Set	5-44
Step 26: (Required) Define BBI-SS PAS Suffixes and Target System	
Parameters	5-46
Step 27: (Optional) Define Configuration Members to BBI-SS PAS ...	5-51

Step 28: (Optional) Specify BBI Internal Services Parameters	5-53
Step 29: (Optional) Set BBI-SS PAS Historical Recording Interval	5-56
Step 30: (Optional) Allocate BBI-SS PAS Journal and Image Logs	5-57
Step 31: (Required) Create a BBI-SS PAS Start Procedure	5-59
Step 32: (Required) Allocate MAINVIEW AutoOPERATOR Data Sets	5-64
Step 33: (Required) Create Automatic BBI-SS PAS Log Handling Procedures	5-65
Step 34: (Optional) Define Terminal Session (TS) Parameters	5-68
Step 35: (Optional) Define Cross-System Communication	5-72
Plan for CAS to CAS Communication	5-74
Plan for BBI-SS PAS to BBI-SS PAS Communication	5-79
Define BBI-SS to BBI-SS PAS Communication	5-82
Define a CAS to CAS Communication	5-86
Step 36: (Required) Specify Product Initialization within a BBI-SS PAS 5-102	
Step 37: (Optional) Enable Automatic Restart Manager Support	5-103
Step 38: (Required) Specify Product Option Password Keys	5-104
Step 39: (Required) Add CAS ID to BBI-SS PAS	5-109
Step 40: (Optional) Add Extended Function Data Sets to BBI-SS PAS StartUp Procedure	5-110
Step 41: (Required) Update TSO Command Processor Table	5-111
Step 42: (Optional) Specify Product Initialization within the MVS PAS	5-113

Appendix A

Library Member for CAS

Appendix B

Library Member for MVS PAS

Appendix C

Library Members for BBI-SS PAS

BBPARM Member BBIBBCFG Parameters	C-2
BBPARM Member BBIISP00 Parameters	C-3
BBPARM Member BBISSPxx Parameters	C-4
BBISSPxx Parameters for the BBI-SS PAS Product Environment	C-4
BBISSPxx Parameters for MAINVIEW AutoOPERATOR Only	C-6
BBISSPxx Parameter for MAINVIEW for CICS, MAINVIEW for IMS Online, MAINVIEW for DBCTL, and MAINVIEW for DB2	C-8
BBISSPxx Parameter for MAINVIEW AutoOPERATOR for MQSeries and MAINVIEW for WebSphere MQ Integrator	C-8
BBPARM Member BBTTCP00 Parameters	C-9
BBPROF Data Set Members	C-10

Appendix D

Library Member for MAINVIEW Alarm Manager PAS

Appendix E

Implementing EXCP or VTAM Terminal Sessions

Using AutoCustomization	E-1
Deactivating Previous VTAM Support	E-2

Index

Figures

Figure 1-1	BBI Architecture	1-2
Figure 3-1	MAINVIEW Product Library Customization	3-3
Figure 5-1	BBAIVP JCL to Validate Product Installation	5-4
Figure 5-2	Selection Command to Add MAINVIEW to an ISPF Menu	5-19
Figure 5-3	Define MAINVIEW Cross-System Communication	5-73
Figure 5-4	MAINVIEW Cross-System Communication in a Sysplex Environment	5-74
Figure 5-5	BBI-SS PAS to BBI-SS PAS Communication	5-82
Figure 5-6	BBI-SS PAS to BBI-SS PAS Single-Domain Communication	5-84
Figure 5-7	BBI-SS PAS to BBI-SS PAS Cross-Domain Communication	5-86
Figure 5-8	Default Mode Table Entry for the RUSIZES Parameter	5-90
Figure 5-9	JCL for Assembling a Mode Table Entry	5-91
Figure 5-10	ADD CAS SYSTEM DEFINITION Panel	5-96
Figure 5-11	VTAM Mode Table Entry for NCP	5-99

Tables

Table 3-1	MAINVIEW Product Library Configuration	3-19
Table 4-1	MAINVIEW Product Groups	4-2
Table 4-2	MAINVIEW Customization Steps	4-3
Table A-1	CAS Started Task Procedure Statements	A-1
Table A-2	CAS PROC Statement Parameters	A-2
Table B-1	MVS PAS Started Task Procedure Statements	B-1
Table B-2	MVS PAS PROC Statement Parameters	B-4
Table C-1	BBPARM Member BBIBBCFG	C-2
Table C-2	BBPARM Member BBIISP00	C-3
Table C-3	BBPARM Member BBISSP _{xx} - BBI-SS PAS Control Parameters . . .	C-4
Table C-4	BBPARM Member BBISSP _{xx} - MAINVIEW AutoOPERATOR Parameters C-6	
Table C-5	BBPARM Member BBISSP _{xx} - CICS, IMS Online, DBCTL, and DB2 Parameter	C-8
Table C-6	BBPARM Member BBISSP _{xx} - MQSeries Parameter	C-8
Table C-7	BBPARM Member BBTTCP00	C-9
Table C-8	BBPROF Data Set	C-10
Table 4-1	MAINVIEW Alarm Manager PAS JCL Statements	D-1
Table 4-2	MAINVIEW Alarm Manager PAS PROC Statement Parameters	D-2

About This Book

This book contains the instructions for setting up the operational environment of all MAINVIEW products to your site's requirements. It can be used to set up your environment manually, or if you use AutoCustomization, as a reference to assist you during this process.

How This Book Is Organized

This book is organized as follows:

- Chapter 1 introduces the MAINVIEW Infrastructure, describes how all its components work together, shows what tasks are to be done to implement the environment for your MAINVIEW products, and shows you where task instructions are described.
- Chapter 2 describes what you need to consider when you implement a new MAINVIEW Infrastructure release and you have a previous release.
- Chapters 3, 4, and 5 provide the instructions for manually customizing the MAINVIEW Infrastructure environment for all MAINVIEW products.
- The appendixes provide supplemental information to help you with MAINVIEW customization.

Conventions

The following conventions are used throughout this book to define command syntax and should not be included with a command:

- Brackets [] enclose optional parameters or keywords
- Braces { } enclose a list of parameters; one must be chosen
- A line | separates alternative options; one can be chosen
- An underlined parameter is the default
- An ITEM IN CAPITAL LETTERS indicates exact characters; usage can be all uppercase or lowercase
- Items in *italicized, lowercase* letters are values that you supply
- Commands in uppercase and lowercase letters, such as HSplit, show the command abbreviation by uppercase letters (HS, for example); lowercase letters complete the entire command name as an optional entry

Commands that do not have an abbreviation are in all upper case letters, such as END.

Command Notations

The following notations are used with MAINVIEW commands:

- A semicolon ; stacks two or more commands:

TRANSFER *target product;view*

where *target* is the system or subsystem being monitored, *product* is the MAINVIEW product monitoring a system or subsystem, and *view* is the name of the view to format performance information for display.

Note: A semicolon is the ISPF default delimiter for command stacking. If you change the default to a different character, the semicolon delimiter for MAINVIEW commands also changes to this character.

-
- A period . directs a command to a window other than the current window:

```
EZALARM;W2.ALARM
```

- Positional qualifiers can be a question mark ? or a plus +; generic qualifiers can be an asterisk *:

```
MVS*
```

- An asterisk * used with the CONTEXT command specifies the current system and with the TIME command specifies the current time frame:

```
TIME * * *
```

This requests a time frame of the current date, time, and duration. For more information about the **TIME** command, enter **HELP TIME** on the **COMMAND** line.

- An asterisk * acts as a place holder for positional parameters used with the **PARm** command. For more information about this command, enter **HELP PARm** on the **COMMAND** line.
- An equals sign = used with the CONTEXT command specifies the context from a previous CONTEXT request and with the TIME command can specify the date, time, or duration from a previous TIME request:

```
TIME 15APR1998 = =
```

This requests the time and duration specified with the previous TIME command.

MAINVIEW Library

The MAINVIEW product family includes the following products:

CMF MONITOR
MAINVIEW AutoOPERATOR
MAINVIEW Explorer
MAINVIEW FOCAL POINT
MAINVIEW for CICS
MAINVIEW for DB2
MAINVIEW for DBCTL
MAINVIEW for IMS Offline
MAINVIEW for IMS Online
MAINVIEW for IP
MAINVIEW for Linux – Servers
MAINVIEW for OS/390
MAINVIEW for UNIX System Services (USS)
MAINVIEW for VTAM
MAINVIEW for WebSphere Application Server
MAINVIEW for WebSphere MQ (formerly known as MAINVIEW for MQSeries)
MAINVIEW Storage Resource Manager (SRM)
MAINVIEW SYSPROG Services
MAINVIEW VistaPoint

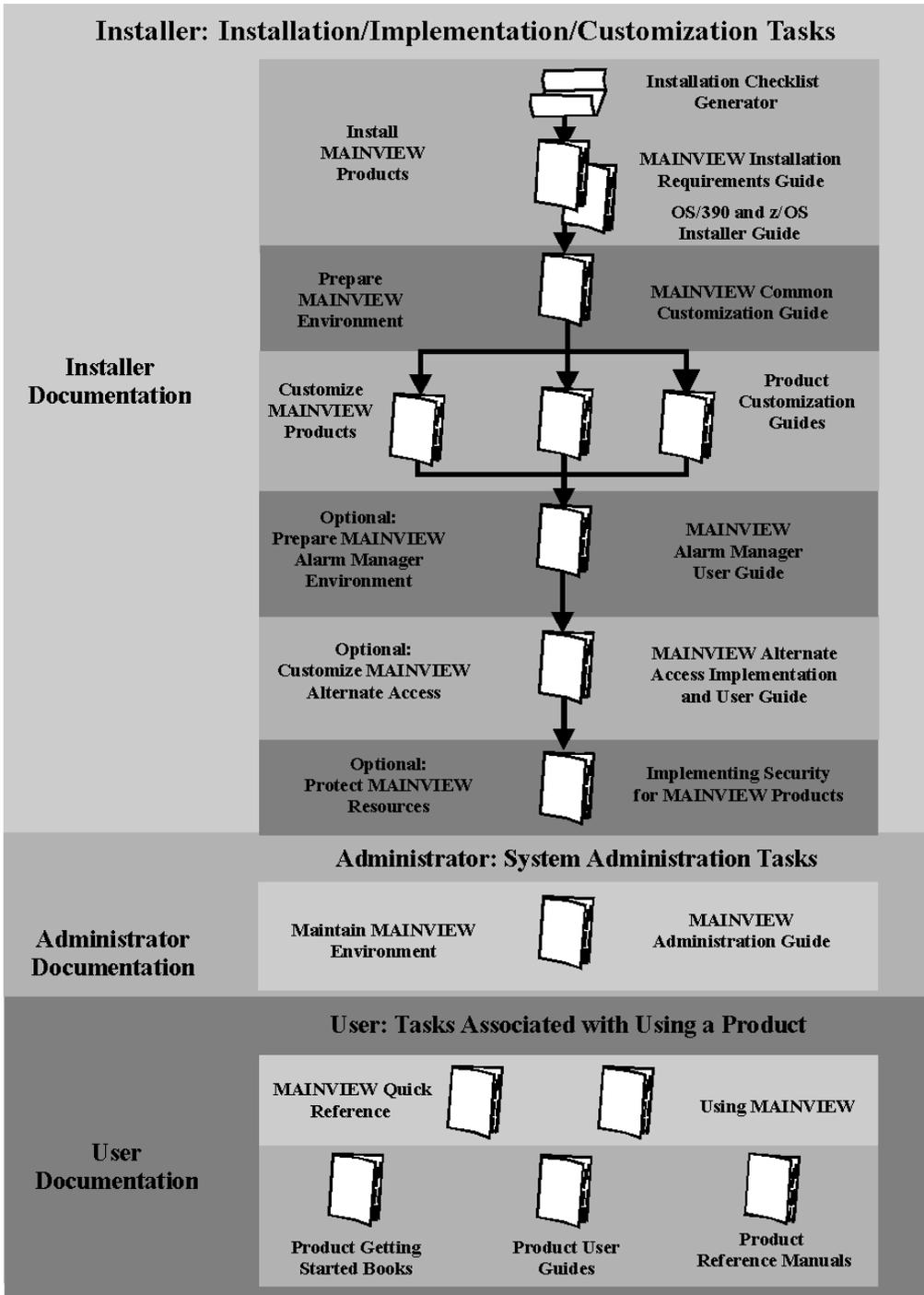
Each of these products provides a product-specific library that typically includes getting started, customization, user, and reference books.

In addition to those books, there are several books and quick references that provide general information common to all or many MAINVIEW products. Those books are listed and described in the following table.

OS/390 and z/OS Installer Guide	provides information about the installation of BMC Software products on OS/390 and z/OS systems.
MAINVIEW Installation Requirements Guide	provides information on installation requirements such as software requirements, storage requirements, and system requirements.
MAINVIEW Common Customization Guide	provides instructions for manually customizing the MAINVIEW environment for your products.
MAINVIEW Alarm Manager User Guide	explains how to create and install alarm definitions that indicate when exceptions occur in a sysplex.
MAINVIEW Alternate Access Implementation and User Guide	explains how to configure, start, and stop VTAM and EXCP AutoLogon sessions to access MAINVIEW products without an active TSO subsystem.
Implementing Security for MAINVIEW Products	explains basic MAINVIEW security, enhanced security, and MAINVIEW Alternate Access security.
MAINVIEW Administration Guide	provides information on MAINVIEW operations, targets, single-system image contexts, MAINVIEW Alarm Manager, data sets, view customization, and diagnostic facilities.
MAINVIEW Quick Reference	introduces the MAINVIEW family of products and lists the commands used to manage the MAINVIEW windows environment.
Using MAINVIEW	provides information on working with MAINVIEW products in windows mode and full-screen mode.

Note: MAINVIEW messages are documented in the Messages and Codes online display, which you can access by typing MSG in the command line of any MAINVIEW display.

The following figure shows the documentation for MAINVIEW products and its intended use.



Online and Printed Books

The books that accompany BMC Software products are available in online and printed formats. Online books are formatted as Portable Document Format (PDF) files. Some online books are also formatted as HTML files.

To Access Online Books

To view any online book that BMC Software offers, visit the Customer Support page of the BMC Software Web site at <http://www.bmc.com/support.html>. You can also access PDF books from the documentation compact disc (CD) that accompanies your product.

Use the free Acrobat Reader from Adobe Systems to view, print, or copy PDF files. In some cases, installing the Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader from the Web, go to the Adobe Systems site at <http://www.adobe.com>.

To Request Additional Printed Books

BMC Software provides some printed books with your product order. To request additional books, go to <http://www.bmc.com/support.html>.

Online Help

MAINVIEW products include extensive online Help. You can access Help by pressing **F1** from any product view or ISPF panel.

Release Notes and Other Notices

Printed release notes accompany each BMC Software product. Release notes provide current information such as

- updates to the installation instructions
- last-minute product information

In addition, BMC Software sometimes provides updated product information between releases (in the form of a flash or a technical bulletin, for example). The latest versions of the release notes and other notices are available on the Web at <http://www.bmc.com/support.html>.

Chapter 1 Customizing MAINVIEW

This chapter discusses two concepts that you need to understand in order to customize your MAINVIEW environment for product operation:

- MAINVIEW Infrastructure
- MAINVIEW product customization

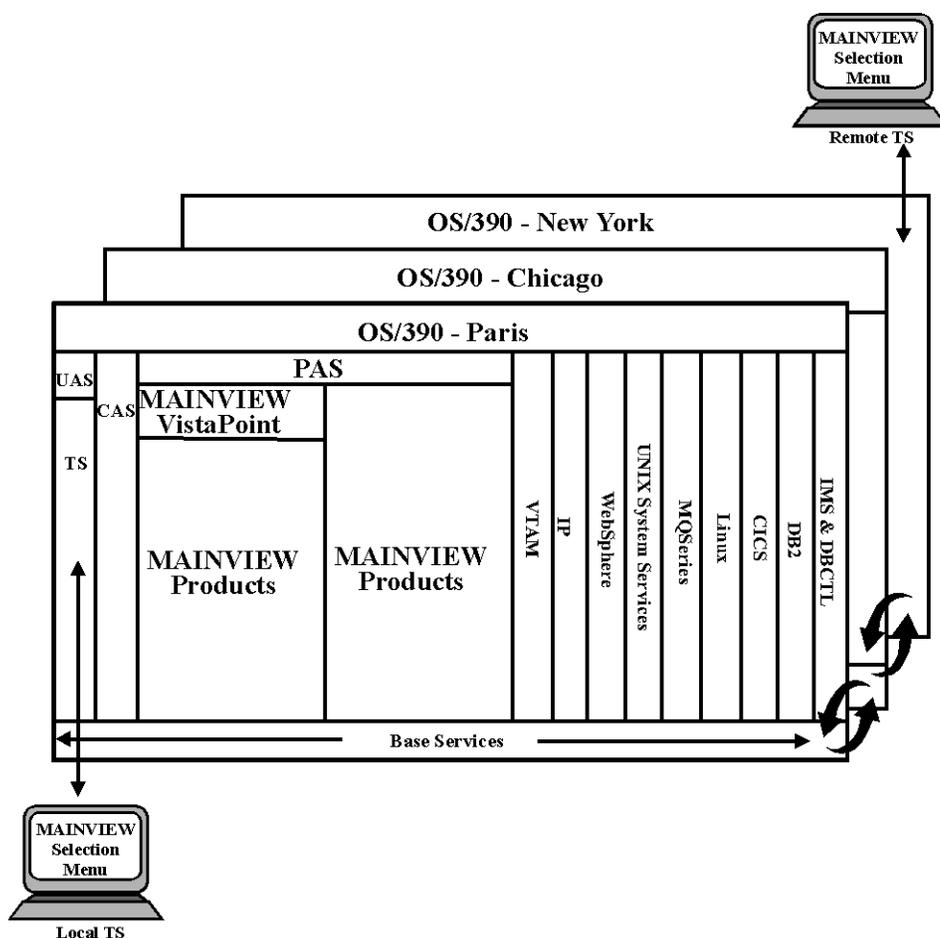
Understanding the MAINVIEW Infrastructure

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 allows 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 integrates the MAINVIEW performance products within a common communications framework that operates across multiple machines in multiple locations as shown in Figure 1-1 on page 1-2. This integrated architecture allows a single terminal session, using one or more MAINVIEW products, to monitor and manage multiple local or remote targets, whether OS/390 itself (sysplex and nonsysplex) or subsystems like CICS, DB2, IMS, IP, Linux, MQSeries, UNIX, VTAM, and WebSphere.

Figure 1-1 BBI Architecture



For products that operate in MAINVIEW windows mode, this 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 MVS systems. Depending on the products installed, this allows you to

- Access different systems and products quickly and easily with simple target switching, direct hyperlinks between products, or multiple concurrent views on one terminal session

- Summarize data on a single system or across multiple MVS images
- View historical or realtime 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 Space (CAS)

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

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 product 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 SSI contexts
- control security for products

Product Address Space (PAS)

The PAS runs as an MVS subsystem. It comprises special routines, including data collectors, to support one or more MAINVIEW products.

- The MVS PAS supports the following MAINVIEW products:
 - CMF MONITOR
 - MAINVIEW for OS/390
 - MAINVIEW for UNIX System Services
 - MAINVIEW SYSPROG Services
 - MAINVIEW VistaPoint (for MVS workloads)

There is one MVS PAS per MVS image. The MVS PAS always connects to the CAS on that MVS image.

- The BBI-SS PAS supports the following MAINVIEW products:
 - MAINVIEW AutoOPERATOR
 - MAINVIEW for CICS
 - MAINVIEW for DB2
 - MAINVIEW for DBCTL
 - MAINVIEW for IMS Online
 - MAINVIEW for WebSphere MQ (formerly known as MAINVIEW for MQSeries)
 - MAINVIEW VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)

Multiple instances of the BBI-SS PAS can run on a single MVS image and contain one or more products. Depending on the products installed, the BBI-SS PAS may or may not connect to a CAS on that MVS image.

BBI-SS PASs on local and remote systems are linked together to provide cross-system communication for an individual terminal session through a local BBI-SS PAS to any other BBI-SS PAS.

- Product-specific PASs support the following MAINVIEW products:
 - MAINVIEW for IP
 - MAINVIEW for Linux – Servers
 - MAINVIEW for VTAM
 - MAINVIEW for WebSphere Application Server
 - MAINVIEW Storage Resource Manager (SRM)

For MAINVIEW for Linux – Servers and MAINVIEW for VTAM, multiple instances of the PAS can run on a single MVS 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 you plan to monitor.

For MAINVIEW for IP, MAINVIEW for WebSphere Application Server, and MAINVIEW SRM, only one product-specific PAS can be active for each product on an MVS image.

- The MAINVIEW Alarm Manager PAS supports all MAINVIEW products that run in windows mode.

The products must be connected to the same CAS as MAINVIEW Alarm Manager. See the *MAINVIEW Alarm Manager User Guide* for more information.

User Address Space (UAS)

The UAS is the home for a terminal session. A terminal session provides the end-user session for all MAINVIEW products. The terminal session connects to a CAS if there is one available, or to a BBI-SS PAS, or to both. There are three types of UASs:

- TSO address space

Using a TSO address space, your terminal session can access MAINVIEW products and perform other TSO/ISPF functions.

- VTAM or EXCP address space using MAINVIEW Alternate Access

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.

- MAINVIEW Host Gateway using MAINVIEW Explorer

Using the MAINVIEW Explorer Web browser, your terminal session can access MAINVIEW products from a PC.

Understanding MAINVIEW Product Customization

This section provides an overview of the steps involved in customizing the MAINVIEW environment for product operation.

MAINVIEW product customization and operation involve the following steps:

1. Install MAINVIEW products

The *OS/390 and z/OS Installer Guide* contains procedures for installing MAINVIEW products. The *MAINVIEW Installation Requirements Guide* lists the software requirements, storage requirements, and system requirements for each product.

2. Prepare the MAINVIEW Environment

This book, *MAINVIEW Common Customization Guide*, contains the procedures for manually customizing the MAINVIEW environment to your site's requirements.

Note: If you used AutoCustomization when you installed your MAINVIEW products, the products are ready for use. AutoCustomization, which is described in the *OS/390 and z/OS Installer Guide*, is an ISPF application that assists you in customizing MAINVIEW products.

3. Customize MAINVIEW Products

Product customization guides are shipped with each MAINVIEW product. A customization guide contains the procedures for tailoring a product and activating additional functions after you have prepared the MAINVIEW environment. If you use AutoCustomization, a product may provide additional AutoCustomization steps.

4. (Optional) Customize MAINVIEW Alarm Manager

The *MAINVIEW Alarm Manager User Guide* contains the procedures for manually implementing, customizing, and using the MAINVIEW Alarm Manager product.

5. (Optional) Customize MAINVIEW Alternate Access

The *MAINVIEW Alternate Access Implementation and User Guide* contains installation, customization, and usage information for EXCP or VTAM communication.

6. Secure MAINVIEW Product Resources

The *Implementing Security for MAINVIEW Products* book contains procedures for securing MAINVIEW product resources with your external security manager (ESM), such as RACF, CA-ACF2, or CA-TOP SECRET.

7. Manage the MAINVIEW Environment

The *MAINVIEW Administration Guide* contains the information you need to administer the MAINVIEW environment, such as cross-system communication, startup procedures for a terminal session or product address space (PAS), historical data set management, and creating site-specific online help.

8. Use a MAINVIEW Product

The *Using MAINVIEW* book contains information about product interfaces and common MAINVIEW functions. Information about specific product functions are in the getting started books, user guides, and reference manuals that support each product.

Chapter 2 Migration Considerations

Version 4.1 is the current version of the MAINVIEW Infrastructure. This chapter applies to users who:

- have MAINVIEW products that are running a version of MAINVIEW Infrastructure prior to version 4.1
- are migrating to MAINVIEW Infrastructure version 4.1

The current MAINVIEW Infrastructure version installed at your site is indicated by *Rvrm* in a CAS startup message, where *v* is the version number, *r* is the release number, and *m* is the modification number as shown in the following example:

```
BMZA001I      CAS(SYSB) SSID(BBCS) INITIALIZATION COMPLETE -  
              V4.1.01 (BPY5615)
```

Migrating CAS Definitions

When you upgrade to a new version of MAINVIEW Infrastructure, you have the option of migrating your existing CAS definitions to the new system. CAS definitions are stored in a member of the *hilevel*.UBBPARM data set. Copying the CAS definitions forward will enable the new CAS to know about all of your existing CASs. You can either

- use AutoCustomization to perform this step when you install MAINVIEW Infrastructure, as described in the *OS/390 and z/OS Installer Guide*
- perform this step manually, as described in “Step 16: (Optional) Migrate CAS Definitions” on page 5-27

Running Mixed Versions of MAINVIEW Infrastructure

For products that run with MAINVIEW Infrastructure, the UAS (where the terminal session resides) and the PAS must be running the same version as the CAS to which they connect. Mixed versions of MAINVIEW Infrastructure can run in a sysplex. If one target system is at a different version than the terminal session on another system, then some functions on the different systems may not be available.

Each new MAINVIEW product version provides its own commands and features plus the commands and features of a previous version. The older versions, however, do not support all the commands and features of the latest versions.

This section describes how you can run mixed versions of MAINVIEW Infrastructure in a sysplex, either on different MVS systems or on the same MVS system.

On Different MVS Systems

If you migrate to a new version of MAINVIEW Infrastructure one MVS system at a time, you will have mixed versions of the infrastructure on different MVS systems in a sysplex.

To implement a new version of MAINVIEW Infrastructure one system at a time, follow these steps:

Step 1 Bring down the existing CAS and all connected PASs on the system being migrated.

Step 2 Copy member BBMTYB00 from your existing UBBPARM data set to the new UBBPARM data set that was allocated during the installation of MAINVIEW Infrastructure.

This step makes all of your existing CAS definitions available to the new CAS that you are about to start.

Step 3 Start the CAS that is running the new version of MAINVIEW Infrastructure.

The following messages are issued at CAS startup:

```
BBMZA096I    CAS(ssid) IS VERSION v.r.mm (mod_id) AND PRIOR START
              WAS VERSION v.r.mm (mod_id)
```

```
BBMZA098I    CAS(ssid) AUTOMATIC COLD START IN PROGRESS FOR
              THIS STARTUP
```

These messages indicate that the connecting CAS is at a different release level than the last CAS that was started. The situation these messages describe is normal since you are in the process of migrating from one version of MAINVIEW Infrastructure to another.

Step 4 Start one or more PASs that are running the new version of MAINVIEW Infrastructure.

A message is issued to indicate that a PAS successfully connected to the new CAS; for example:

```
BBDDI029I    OS/390 PAS(ssid) INITIALIZATION COMPLETE - (mod_id)
```

Step 5 Start one or more UASs that are running the new version of MAINVIEW Infrastructure.

If you attempt to connect a UAS (where the terminal session resides) that is running one version of MAINVIEW Infrastructure to a CAS that is running a different version, the following messages are issued:

```
BBMZA103I    CAS(ssid) IS VERSION v.r.mm (mod_id) AND CLIST VERSION
              IS v.r.mm (mod_id)
```

```
BBMZA102I    SELECT A CAS THAT MATCHES YOUR CLIST RELEASE
              OR EXIST CLIST AND RETRY
```

This message indicates that the CLIST being used for the terminal session tried to connect to a CAS that is running a different version of MAINVIEW Infrastructure than the CLIST libraries. Remember that the UAS must be running the same version as the CAS to which it connects.

On the Same MVS System

If you are testing new MAINVIEW product releases on the same MVS system as existing products, you may have mixed versions of MAINVIEW Infrastructure on the same MVS system. Ideally, a separate MVS system should be used for testing. Sometimes, however, you may want to test a new version on an MVS system that currently runs a CAS and you may want to keep that CAS available to end users. You can do this without disturbing your current environment by defining a separate CAS, PAS, and terminal session for each version of MAINVIEW Infrastructure.

If you have a single MVS system with mixed versions of MAINVIEW Infrastructure for testing and production, the following conditions must be met:

- The test system must have a different terminal session, PAS, and CAS than the production system.
- Each CAS must have a unique subsystem ID (SSID).
- The MAINVIEW CLIST from a previous version of MAINVIEW Infrastructure must have a different name than the current MAINVIEW CLIST.

To implement a new version of MAINVIEW Infrastructure on the same system as a previous version, follow these steps:

Step 1 Copy member BBMTYB00 from your existing UBBPARM data set to the new UBBPARM data set that was allocated during the installation of MAINVIEW Infrastructure.

This step makes all of your existing CAS definitions available to the new CAS that you are about to start.

Step 2 Start the CAS that is running the new version of MAINVIEW Infrastructure. Be sure to specify

- a unique CAS name on the SSID parameter (by default, the CAS name is the MVS system name)
- a unique XCF group name by using the DFLTGRP parameter

The following CAS startup JCL illustrates these parameters:

```
//C410 PROC SSID=C410
//      COLD=N
//      CONVXCF=N
//      DFLTGRP=GROUPF
//      DUMP=Y
//      EMM=N
//      XDM=N
//CAS   EXEC PGM=BBM9ZA00 ,
//      PARM=( 'SSID=&SSID , XDM=&XDM , COLD=&COLD ' ,
//      ' DUMP=&DUMP , EMM=&EMM , DFLTGRP=&DFLTGRP ' ,
//      ' CONVXCF=&CONVXCF ' ) ,
//      TIME=1439 ,
//      REGION=4096K
//*
```

For complete information about the CAS startup parameters, refer to the *MAINVIEW Administration Guide*.

Step 3 Start one or more PASs that are running the new version of MAINVIEW Infrastructure.

A message is issued to indicate that a PAS successfully connected to the new CAS; for example:

```
BBDDI029I      OS/390 PAS(ssid) INITIALIZATION COMPLETE - (mod_id)
```

Step 4 Start one or more UASs that are running the new version of MAINVIEW Infrastructure.

The CLIST for the new version must have a different name and allocate different user libraries than previous versions of MAINVIEW Infrastructure. To access products at different version levels on the same system, you can exit and re-enter ISPF, then run the appropriate MAINVIEW CLIST.

Step 5 Use the Plex Manager CASDEF view to add one of the following to the definition of the CAS you just started:

- a VTAM APPLID for VTAM communications
- a TCP/IP address and port number for TCP/IP communications

Once you stop and restart the new CAS, the new communications parameters will be used.

Chapter 3 Using MAINVIEW Product Libraries

Of the distributed target libraries, only those libraries and data sets that are changed by customization are described in this section. The purpose of this section is to ensure that site changes to customized product libraries are not lost when your site migrates to a new version or applies product maintenance.

This section provides information about which product libraries to use when you make changes to a product. It explains how the product libraries are created, what their intended use is, and which libraries to use to make your site's changes.

The types of product libraries are:

- SMP-maintained distributed target libraries

These are created during product installation. They contain load modules, parameters, procedures, samples, views, screen definitions for views, and view help in their original form. The installation procedures are described in the *OS/390 and z/OS Installer Guide*.

Note: Never modify these libraries without SMP procedures.

- Site-customized product libraries

These are created for you by AutoCustomization, or you can create them manually. They contain versions of distributed library members modified to your site's requirements.

- Product user libraries

Each user can have their own version of views, screen definitions containing views, and view help in a library with their user ID.

A user profile (*uprefix.userid.BBPROF*) is created during terminal session initialization if one does not exist already, depending on what products are installed.

- Image and Journal Logs

These are used by some products that run in the BBI-SS PAS for recording screen images or messages. They are created for you by AutoCustomization, or you can create them manually.

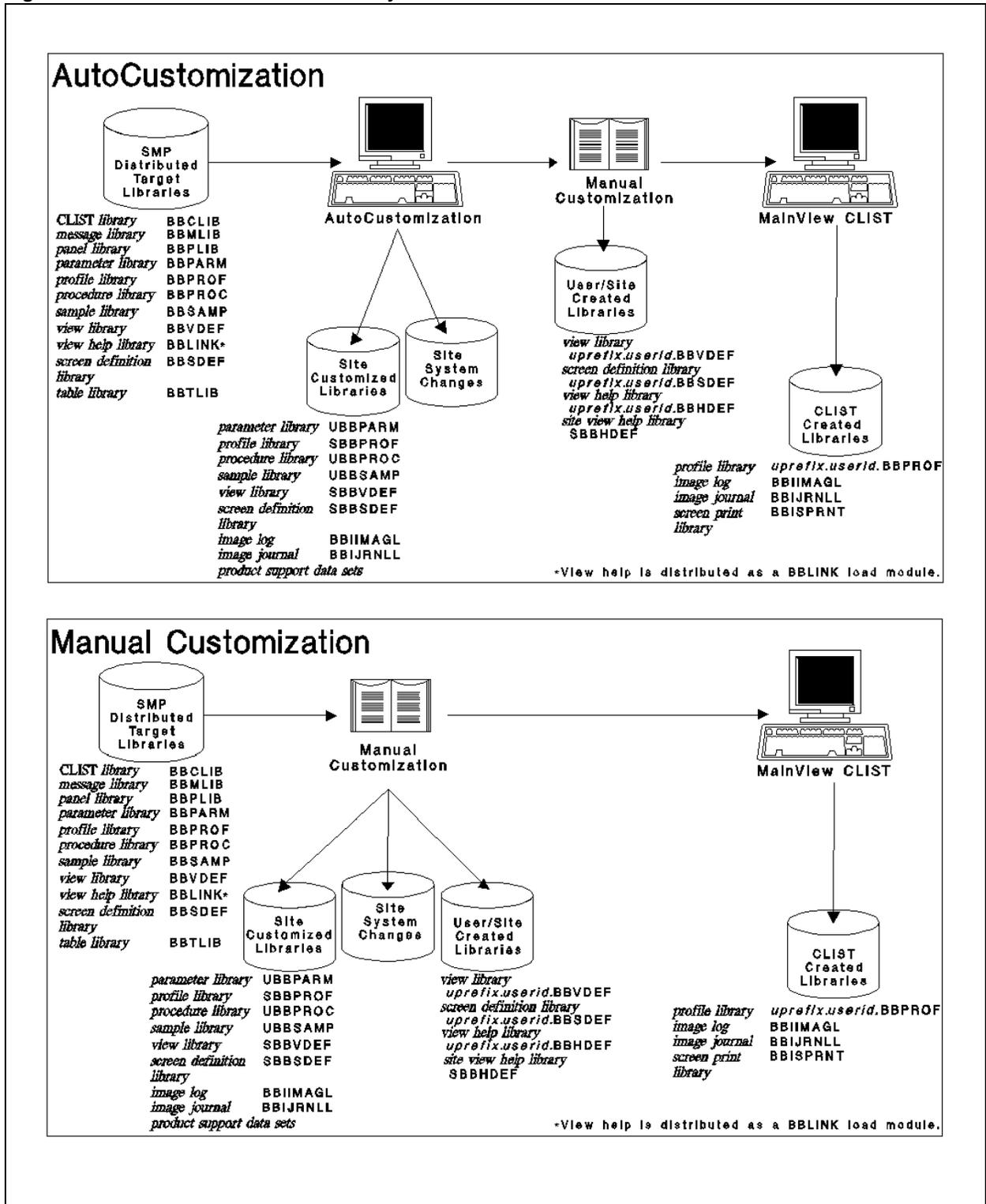
- Product support data sets

These data sets are required for some product functions. They are created for you by AutoCustomization, or you can create them manually.

Note: AutoCustomization procedures are described in the *OS/390 and z/OS Installer Guide*. Manual procedures for creating product libraries are described in the *MAINVIEW Common Customization Guide*.

Figure 3-1 on page 3-3 shows how customized product libraries are created.

Figure 3-1 MAINVIEW Product Library Customization



Distributed Libraries

The distributed libraries are as follows:

Link library This data set contains MAINVIEW environment and product-specific load modules.

Parameter library
Product parameters

Members in this library contain parameter definitions for related products and product-specific parameter definitions.

Sample library Product samples

Members in this library contain macros, sample JCL, and sample user exit routines.

Profile library Profile information

This library contains members that define PF keys, target system defaults, primary option menu, unique application profiles, and cycle refresh definitions for a user's terminal session for the products that run in full-screen mode.

Procedure library
MAINVIEW AutoOPERATOR execute command lists (EXECs)

Members in this library contain user-defined automated applications.

View library Product views

Members in this library contain views used by products operating in windows mode.

Screen library Screen definitions with views

Members in this library include screen definitions containing one or more views.

View help library
Help text

This library contains help text for products providing views.

Message library Messages and abend codes

This library, called BBMLIB, contains messages and abend codes issued by MAINVIEW products.

Panel library ISPF panels

Each member in this library, called BBPLIB, is a panel definition for the terminal session. User-customized panels should be placed in a separate library and concatenated in front of the distributed panel library.

CLIST library MAINVIEW CLISTs

AutoCustomization and CLISTs used by specific MAINVIEW products are in this library, called BBCLIB.

Table library MAINVIEW tables

This library, called BBTLIB, contains MAINVIEW product tables, AutoCustomization tables, and some MAINVIEW product command tables.

Note: Use the contents of the distributed libraries as models to create your own site-customized product libraries. The distributed libraries should never be modified. All changes described in this manual should be made only to site-customized product libraries. If you change the distributed libraries, subsequent SMP maintenance will overwrite your changes.

Customized Libraries

The customized libraries include some or all of the following:

- your own versions of the distributed libraries
- product support data sets

These include the following:

— BBIBBCFG

This data set is used by MAINVIEW AutoOPERATOR, MAINVIEW FOCAL POINT, MAINVIEW for CICS, MAINVIEW for DB2, MAINVIEW for DBCTL, MAINVIEW for IMS Online, MAINVIEW for WebSphere MQ, and MAINVIEW VistaPoint. BBIBBCFG identifies BBPARM configuration member suffixes to the BBI-SS PAS.

— BBIDIV, BBIVARS

These data sets are used by MAINVIEW AutoOPERATOR products. BBIDIV is a data storage data set. BBIVARS is a variable pool data set.

— PARMFILE

This data set is for MAINVIEW for OS/390 workload configuration definitions and for products supporting historical data processing with the view TIME command.

- image and journal logs

These are used by MAINVIEW AutoOPERATOR, MAINVIEW for CICS, MAINVIEW for DB2, MAINVIEW for DBCTL, and MAINVIEW for IMS Online. The Image log is for screen images from timer-driven service requests. The Journal log is for BBI-SS PAS and product commands, responses, and messages.

Parameter Libraries

Each member in this data set contains parameters for a separate product or group of related products. Your site can have several parameter libraries, a distributed version and one or more site-customized versions. Multiple parameter library data sets can be concatenated together. With this technique, a site parameter library set can be created and a data set individualized for each PAS can be concatenated before the site library.

BBPARM

BBPARM is a distributed target library that is created during product installation. This version should never be modified. Subsequent SMP maintenance will overwrite any changes you make. The distributed name is used in this manual for reference only.

UBBPARM

UBBPARM is a copy of the distributed parameter library. It can be created automatically by AutoCustomization, or you can create it manually.

Note: AutoCustomization creates a UBBPARM data set as *hilevel.image.UBBPARM*; for example, *hilevel.IMAGSYSD.UBBPARM*. This lets each MVS image have its own BBPARM data set for unique parameters.

Use this copy to make any parameter library changes described in this manual.

AutoCustomization

If you used AutoCustomization when you installed your product libraries as described in the *OS/390 and z/OS Installer Guide*, it created a UBBPARM data set for you and customized its members.

Manual Customization

If you are customizing the product manually:

1. Create your own UBBPARM data set.
2. Copy the members that you need from the distributed BBPARM library into the UBBPARM data set that you created.
3. Change the members that you copied in UBBPARM.

Customized IMS Parameters

You can define unique parameters for individual IMS systems, either by copying and renaming specific members and including them in UBBPARM or by creating user-defined parameter data sets, referred to as *ibbparm* data sets.

These methods for customizing parameters are used only by MAINVIEW for IMS Online, MAINVIEW for IMS Offline, MAINVIEW for DBCTL, and MAINVIEW AutoOPERATOR for IMS.

Customized IMFSYS00 and IMFECPO0 Members

In the UBBPARM data set, you can include a copied, renamed version of IMFSYS00 to customize the system parameters for an IMS system, including the SUBSYS parameter, which establishes communication between the IMS and the BBI-SS PAS. The name of an IMS-specific system parameter member must be in the following format:

imsidSYS (where *imsid* is the four-character IMS identification code)

You can also include a copied, renamed version of IMFECPO0 to customize the Event Collector parameters for an IMS system. The name of an IMS-specific Event Collector parameter member must be in the following format:

imsidECP (where *imsid* is the four-character IMS identification code)

If you do not need to customize other UBBPARM members for an IMS system, you do not need to create and allocate a separate *ibbparm* parameter data set for that system.

User-Defined Parameter Data Set

You can allocate an *ibbparm* parameter data set to contain any members that you want to make unique to an IMS system. The *ibbparm* data set must be allocated manually; it is not created through AutoCustomization.

If you do not need to customize any parameter members other than IMFSYS00, IMFSYS00, or both for an IMS system, an *ibbparm* data set is not required. Instead, you can include renamed versions of the members in UBBPARM, as described in “Customized IMFSYS00 and IMFECPO0 Members” on page 3-7.

Sample Libraries

The members in this data set contain:

- sample JCL that can be edited and submitted to perform specified functions
- macros that are referenced when assembling user-written services
- sample user exit routines

Your site can have several sample libraries, a distributed version and one or more site-customized versions. Some members are for more than one product and some are product-specific.

BBSAMP

BBSAMP is a distributed target library that is created during product installation. This version should never be modified. Subsequent SMP maintenance will overwrite any changes you make. The distributed name is used in this manual for reference only.

UBBSAMP

UBBSAMP contains copies of members from the distributed sample library. It can be created automatically by AutoCustomization, or you can create it manually. You can use UBBSAMP to make any changes to members described in this manual.

AutoCustomization

If you used AutoCustomization when you installed your product libraries as described in the *OS/390 and z/OS Installer Guide*, it created a UBBSAMP data set for you if one did not exist already. UBBSAMP contains copies of members from the distributed sample library. Use these members to customize a product to your site's needs.

Manual Customization

If you are customizing the product manually:

1. Create your own UBBSAMP data set.
2. To change a sample member described in this manual, copy the members that you need from the distributed BBSAMP library into the UBBSAMP data set that you created.
3. Change the members that you copied in UBBSAMP.

Profile Libraries

This section applies only to:

MAINVIEW AutoOPERATOR
MAINVIEW for CICS
MAINVIEW for DB2
MAINVIEW for DBCTL
MAINVIEW for IMS Online

Your site can have several profile libraries, a distributed version and one or more site-customized versions. Members in this data set contain profile information and cycle refresh definitions. Other members are dynamically created. Do not change any members in this library unless instructed to.

You can have a site profile library and a user profile library. The site library can be created automatically by AutoCustomization, or you can create it manually. The site library is a common profile shared by all site users. The MAINVIEW CLIST creates a user profile automatically if one does not exist already. Users should have their own profile library so that each user can specify:

- unique PF keys
- CYCLE commands
- target system defaults
- primary Option Menu
- a unique set of application profiles

The user profile and the site profile should be concatenated before the distributed profile. When a profile is saved, it is stored in the first profile library defined in the concatenation.

BBPROF

BBPROF is a distributed target library that is created during product installation. This version should never be modified. Subsequent SMP maintenance will overwrite any changes you make. The distributed name is used in this manual for reference only.

SBBPROF

SBBPROF is an optional data set. It can be created automatically by AutoCustomization, or you can create it manually. Use SBBPROF to make any changes described in this manual that you want to be shared by all users at your site.

AutoCustomization

If you used AutoCustomization when you installed your product libraries as described in the *OS/390 and z/OS Installer Guide*, it created an SBBPROF data set for you and customized its members.

Manual Customization

If you are customizing the product manually:

1. Create a common SBBPROF data set for your site.
2. Copy the members that you need from the distributed BBPROF library into the SBBPROF data set that you created.
3. Change the members that you copied in SBBPROF.

User BBPROF

There should be a profile data set for each user so that each user has an individual application profile. The MAINVIEW CLIST created a user profile automatically if one did not exist. It is called *userid.BBPROF*, where *userid* is the user's logon ID. This data set contains profile members customized by a user.

Procedure Libraries (MAINVIEW AutoOPERATOR Only)

Your site can have several procedure libraries available, a distributed version and one or more site-customized versions. Members in this data set contain executable procedures used by MAINVIEW AutoOPERATOR. These procedures are execute command lists (EXECs) that automate site functions. For more information about EXECs, see the manuals shipped with your MAINVIEW AutoOPERATOR product.

BBPROC

BBPROC is a distributed target library that is available when MAINVIEW AutoOPERATOR is installed successfully. This version should never be modified. Subsequent SMP maintenance will overwrite any changes you make. The distributed name is used in this manual for reference only.

UBBPROC

UBBPROC is used to contain new user-written EXECs or customized MAINVIEW AutoOPERATOR-supplied EXECs from the distributed BBPROC library. It can be created automatically by AutoCustomization or you can create it manually.

AutoCustomization

If you used AutoCustomization when you installed MAINVIEW AutoOPERATOR as described in the *OS/390 and z/OS Installer Guide*, it created a UBBPROC data set for MAINVIEW AutoOPERATOR.

If you need to use a specific EXEC sample:

1. Verify that the member was not copied by AutoCustomization to UBBPROC.
2. Copy the sample member that you need to modify from the distributed BBPROC to UBBPROC.
3. Make the change in the copied member.

Manual Customization

If you are customizing MAINVIEW AutoOPERATOR manually:

1. Create your own UBBPROC data set.
2. Copy the members that you need to modify from the distributed procedure library into the UBBPROC data set that you created.
3. Change the members that you copied in UBBPROC.

View, Screen, and View Help Distribution Libraries

This section applies only to those MAINVIEW products operating in windows mode. The distribution libraries for those products are allocated as follows:

Views

The SMP target view library is allocated in the CAS and PAS startup procedure. Site and user view libraries are allocated to the UAS, as described in “Site Libraries” on page 3-13 and “User Libraries” on page 3-14.

This library contains a uniquely named set of views for a product. View tables allow multiple products to have the same view names. For example, MAINVIEW for OS/390 has one version of VIEWS while CMF MONITOR has a completely different version of VIEWS; they are kept separate in the library by being stored in tables with unique names.

Screen Definitions

A set of screen definitions is allocated by the MAINVIEW CLIST.

Help text

Help text for views is distributed as a load module in the *hilevel.BBLINK* load library.

Site Libraries

The following site libraries are created for you during AutoCustomization:

- a view library, allocated as *hilevel.SBBVDEF*
- a screen definition library, allocated as *hilevel.SBBSDEF*

The first S in SBBVDEF and SBBSDEF represents ”site.”

When the MAINVIEW CLIST (created by AutoCustomization) is used to access the MAINVIEW Selection Menu, *hilevel.SBBVDEF* and *hilevel.SBBSDEF* data sets are allocated to a UAS using the DD names: *BBVDEF* and *BBSDEF*.

Note: A system administrator should retain sole authority for adding and deleting views, help text, or screen definitions from site libraries. Granting write authority to more than one user may result in confusion and subject valuable data to overwriting. Users who want to contribute to a site library should contact the system administrator.

If you have an external security system such as RACF, CA-ACF2, or CA-TOP SECRET, it can be used to restrict changes to site libraries from all user IDs except the system administrator's.

User Libraries

User libraries contain customized views, screen definitions, or help text that are available to an individual user only.

Note: If a user does not have access to a user library, any changes made and saved by that user are stored in the site library by default unless the system administrator has prevented write access to the site library. A user library should be created for each user at your site.

1. Create standard partitioned data sets (fixed block, LRECL=80) for a user view, screen definition, and help text library.

Use the following naming convention:

uprefix.userid.BBxDEF

where:

uprefix Is a user's TSO prefix

Note: You can use the UPREFIX parameter in the MAINVIEW CLIST to specify any prefix you want. This is useful for those who have more than one TSO ID and want to use the same user *BBxDEF* data set with all their IDs. If UPREFIX is not specified, the default is *uprefix.userid.BBxDEF*.

userid Is a user's TSO ID

xIs one of the following:

V	View library
S	Screen definition library
H	Help text library

2. Change the user's allocations to contain user and site data sets. Ensure that the user library is searched before the site library.

Note: If the MAINVIEW CLIST is used to access the MAINVIEW Selection Menu, it concatenates the view and screen libraries. The user help text library must be added manually to the concatenation.

As many data sets as needed can be added to the concatenation within your site's restrictions. For example, you may want to create department- or group-specific libraries that only a select group of people may access. All of the people in that department or group, then, would name the same data sets on their BBxDEF concatenation.

Shared Libraries

User libraries allocated to a UAS are shared by all products providing views. Therefore, if you have more than one of these products installed on your system, you must use caution when saving views and screen definitions.

For example, suppose you create a MAINVIEW for OS/390 view called MYVIEW and save it in your user view library. You then access CMF MONITOR and create another view, also called MYVIEW. When you try to save the second MYVIEW, a message warns you that MYVIEW already exists because you created MAINVIEW for OS/390 MYVIEW previously. If you save the second version anyway, the first version of MYVIEW is overwritten.

To avoid possible confusion, you should establish a naming convention at both the site and user levels; for example, Cxxxxxx for CMF MONITOR views, Mxxxxxx for MAINVIEW for OS/390 views.

Library Concatenation

By default, when views, screen definitions, or help text are changed or new ones created and saved, they are saved in the libraries as follows:

1. User library first, if one exists
2. Site library, if one exists

If neither exist, an error message is displayed.

Note: The distributed view library cannot be overwritten because it is allocated to a PAS, not a UAS.

When views, screen definitions, or help are requested, the libraries are searched by default as follows:

1. Your user library first, if one exists
2. Your site library, if one exists
3. The distributed library (for views and help text only)

Link Library

This data set must be authorized. Executable modules are obtained from this library if it is specified. If it is not specified, an error message is generated. This data set is concatenated in the target's STEPLIB DD statements for some MAINVIEW products.

Product Support Data Sets

MAINVIEW AutoOPERATOR, MAINVIEW FOCAL POINT, MAINVIEW for CICS, MAINVIEW for DB2, MAINVIEW for DBCTL, MAINVIEW for IMS Online, MAINVIEW for WebSphere MQ, and MAINVIEW VistaPoint use:

- BBIBBCFG

This BBPARM member specifies which configuration members in the BBPARM library are used when a product starts and connects to the BBI-SS PAS. Configuration members control the way the BBI-SS PAS operates.

MAINVIEW AutoOPERATOR and MAINVIEW Alarm Manager use:

- **BBIDIV**

This data set is required for MAINVIEW AutoOPERATOR and MAINVIEW Alarm Manager. It is used to store binary large objects and data across product restarts. It is allocated in the PAS startup procedure.

MAINVIEW AutoOPERATOR uses:

- **BBIVARS**

A profile variable pool. Variables are written to this data set when an EXEC that issues a VPUT ... PROFILE ends or when the VCKP command is issued. It is allocated in the BBI-SS PAS log procedure.

MAINVIEW for OS/390 and products supporting the DSLIST view and the TIME command use:

- **PARMFILE**

This VSAM data set is for:

- user-defined workload definitions created by the MAINVIEW for OS/390 product
- products providing views of data from a time interval in the past

Data from the past is recorded in historical data sets. PARMFILE is used to contain a directory of the historical data set names, which is shown by a product having a DSLIST view. Data from a historical data set is shown in any of that product's views with the TIME command.

PARMFILE is allocated in the BBI-SS PAS and MVS PAS startup procedures.

Image and Journal Logs

There are two image and two journal logs allocated in the BBI-SS PAS startup procedure that are used by the following products:

- MAINVIEW AutoOPERATOR (only uses the journal logs)
- MAINVIEW Alarm Manager (only uses the journal logs)
- MAINVIEW for CICS
- MAINVIEW for DB2
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for WebSphere MQ (formerly known as MAINVIEW for MQSeries)

The BBI-SS PAS image log records screen images that are produced automatically by timer-driven analyzer and monitor services. Image logging can be disabled by removing the appropriate statement from the BBI-SS PAS startup procedure.

The BBI-SS PAS journal log records:

- all commands and responses issued from a terminal session 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
- BBI-SS PAS informational, error, and audit messages
- service commands and messages
- DB2 commands and messages

Although it is not recommended, because all operational and diagnostic messages are written to the BBI-SS PAS journal log, BBI-SS PAS journal logging can be disabled by removing appropriate DD statements from the BBI-SS PAS startup procedure JCL. BBI-SS PASs cannot share journal data sets.

Note: Image and journal logs and a screen print data set are allocated to a user's terminal session by the MAINVIEW CLIST.

Product Library Customization Summary

The following table summarizes how the MAINVIEW product libraries should be configured.

Table 3-1 MAINVIEW Product Library Configuration

Distributed Target Library	Customized Library	Created by AutoCustomization?	Allocated to which Address Space?	Allocated to What DD Name?
BBLINK	BBLINK	No, AutoCustomization APF-authorizes the distributed BBLINK load library and adds it to your system link library. See "Notes" at the end of the table.	UAS, PAS, CICS target, IMS target	BBLOAD or BBILINK
BBSAMP	UBBSAMP	Yes, AutoCustomization creates a UBBSAMP sample library.	None	None
BBPARM	UBBPARM	Yes, AutoCustomization creates a UBBPARM parameter library. See "Notes" at the end of the table.	PAS, CICS target, IMS target	PARMLIB or BBIPARM
	ibbparm	No, <i>ibbparm</i> is a user-defined parameter library that must be created manually.	IMS target	IMFPARM
BBPARM member BBIBBCFG	UBBPARM member CFGssidA	Yes, AutoCustomization creates the UBBPARM member.	BBI-SS PAS	BBCFG
BBPROC	UBBPROC	Yes, AutoCustomization creates a UBBPROC procedure library but only for MAINVIEW AutoOPERATOR.	BBI-SS PAS	SYSPROC
None	BBIDIV data set	Yes, AutoCustomization creates this data set for MAINVIEW AutoOPERATOR and MAINVIEW Alarm Manager.	PAS	BBIDIV
None	BBIVARS data set	Yes, AutoCustomization creates this data set but only for MAINVIEW AutoOPERATOR.	BBI-SS PAS	BBIVARS

Table 3-1 MAINVIEW Product Library Configuration (continued)

Distributed Target Library	Customized Library	Created by AutoCustomization?	Allocated to which Address Space?	Allocated to What DD Name?
None	BBIIMAG1, BBIIMAG2	Yes, AutoCustomization creates dual image log data sets for: MAINVIEW for: CICS DB2 DBCTL IMS Online See "Notes" at the end of the table.	BBI-SS PAS	BBIIMAG1, BBIIMAG2
None	BBIIJRN1, BBIIJRN2	Yes, AutoCustomization creates dual journal log data sets for: MAINVIEW Alarm Manager MAINVIEW AutoOPERATOR MAINVIEW for: CICS DB2 DBCTL IMS Online MQSeries	PAS	BBIIJRN1, BBIIJRN2
None	BBIIMAGL, BBIJRNLL	No, AutoCustomization does not create an Image or Journal log. They are created by the TSO MAINVIEW CLIST.	UAS	BBIIMAGL BBIJRNLL
BBPROF	<i>uprefix.userid</i> .BBPROF or SBBPROF (or both)	<i>uprefix.userid</i> .BBPROF: No, AutoCustomization does not create a user profile. It is created by the TSO MAINVIEW CLIST. SBBPROF: Yes, AutoCustomization creates a site profile.	UAS	BBIPROF
BBVDEF	<i>uprefix.userid</i> .BBVDEF	No, <i>uprefix.userid</i> .BBVDEF is a user-defined view library that must be created manually. The TSO MAINVIEW CLIST allocates it.	UAS	BBVDEF
	<i>hilevel</i> .SBBVDEF	Yes, AutoCustomization creates a <i>hilevel</i> .SBBVDEF site view library. The TSO MAINVIEW CLIST allocates it.		
	<i>hilevel</i> .BBVDEF	No, AutoCustomization does not create or modify it. It is a distributed library.	PAS	

Table 3-1 MAINVIEW Product Library Configuration (continued)

Distributed Target Library	Customized Library	Created by AutoCustomization?	Allocated to which Address Space?	Allocated to What DD Name?
BBSDEF	<i>uprefix.userid.BBSDEF</i>	No, <i>uprefix.userid.BBSDEF</i> is a user-defined screen definition library that must be created manually. The TSO MAINVIEW CLIST allocates it.	UAS	BBSDEF
	<i>hilevel.SBBSDEF</i>	Yes, AutoCustomization creates a <i>hilevel.SBBSDEF</i> site screen definition library. The TSO MAINVIEW CLIST allocates it.		
	<i>hilevel.BBSDEF</i>	No, AutoCustomization does not create or modify it. It is a distributed library.		
View help text	<i>uprefix.userid.BBHDEF</i>	No, <i>uprefix.userid.BBHDEF</i> is a user-defined help library that must be created manually. Modify the user's TSO logon procedure to allocate it. See "Notes" below.	UAS	BBHDEF
	<i>hilevel.SBBHDEF</i>	No, AutoCustomization does not create a <i>hilevel.SBBHDEF</i> site help library. You must create it manually. The MAINVIEW CLIST allocates it.		
<p>Notes:</p> <ul style="list-style-type: none"> • Concatenation User and site-customized libraries should be concatenated <i>before</i> the distributed libraries. • BBLINK IMS and CICS target allocation of the link library require IMS and CICS JCL modifications. UAS allocation of the link library uses a DD name of BBILOAD and BBILINK. • BBPARM IMS and CICS target allocation of the parameter library require IMS and CICS JCL modifications. • View Help text Help text for views is distributed as a BBLINK load module. 				

Chapter 4 Preparing for Customization

This chapter provides two tables to help you set up your MAINVIEW product environment. The first table, Table 4-1 on page 4-2, categorizes products into groups. Customization and maintenance of the MAINVIEW product environment is the same for all the products in each group. This product grouping is referenced in the customization instructions in Chapter 5, “Customizing the MAINVIEW Environment” and the maintenance instructions in the *MAINVIEW Administration Guide*.

Each step in Chapter 5, “Customizing the MAINVIEW Environment” lists the product group(s) to which that step applies. By looking at Table 4-1 on page 4-2, you can determine whether or not you need to follow those instructions for the MAINVIEW products installed at your site. Some instructions are optional and some are required.

The second table, Table 4-2 on page 4-3, is a list of the customization steps with references to the instructions for each step. These steps parallel the AutoCustomization steps as much as possible. You can refer to them as you use AutoCustomization. The product combination at your site determines the steps that are used. The title for each step described in this book matches the title of the AutoCustomization step as much as possible for easy reference. AutoCustomization, described in the *OS/390 and z/OS Installer Guide*, contains a set of automated dialogs. These dialogs prompt you through customization of the MAINVIEW environment for the product combination installed at your site.

Table 4-1 is the product group table. Refer to this table to determine if you need to follow the instructions for a step described in Chapter 5, “Customizing the MAINVIEW Environment.”

Table 4-1 MAINVIEW Product Groups

Group	Product
All	All products
Group 1	MAINVIEW AutoOPERATOR
	MAINVIEW FOCAL POINT
Group 2	CMF MONITOR
	MAINVIEW for OS/390
	MAINVIEW for UNIX System Services (USS)
	MAINVIEW VistaPoint
	MAINVIEW SYSPROG Services
Group 3	MAINVIEW for CICS
	MAINVIEW for DB2
	MAINVIEW for DBCTL
	MAINVIEW for IMS Online
	MAINVIEW for WebSphere MQ (formerly known as MAINVIEW for MQSeries)
Group 4	MAINVIEW for IP
	MAINVIEW for Linux – Servers
	MAINVIEW for VTAM
	MAINVIEW for WebSphere Application Server
	MAINVIEW Storage Resource Manager (SRM)

Table 4-2 on page 4-3 shows all the tasks for customizing the MAINVIEW product environment with references to the task instructions. An upper-case X character in the table indicates a required step; a lower-case o character indicates an optional step. For instructions about how to tailor a specific product, see the customization guides shipped with your products.

Table 4-2 MAINVIEW Customization Steps

Customization Checklist	MAINVIEW Product Group				
	All	1	2	3	4
"Step 1: (Required) Specify Jobcards and Other Operational Defaults" on page 5-2	X				
"Step 2: (Optional) Review Virtual Storage Requirements" on page 5-3	o				
"Step 3: (Optional) Generate JCL to Run Installation Verification Program" on page 5-4	o				
"Step 4: (Optional) Implement GDDM/PGF Support" on page 5-6			o	o	
"Step 5: (Optional) Implement Katakana Terminal Support" on page 5-7	o				
"Step 6: (Required) Apply Passwords with the Product Authorization Utility" on page 5-9			X	X	X
"Step 7: (Required) Create Site Data Sets for Use with MAINVIEW Products" on page 5-10	X				
"Step 8: (Required for Security) Create Site Security Data Set" on page 5-13			X	X	X
"Step 9: (Required) APF-Authorizing the BMC Software Load Library" on page 5-14	X				
"Step 10: (Optional) Add the BMC Software Load Library to your System Link List" on page 5-16	o				
"Step 11: (Required) Create CLIST for Invoking MAINVIEW Products" on page 5-18	X				
"Step 12: (Optional) Add TSF Authorization for Products to Run Under ISPF" on page 5-20		o		o	
"Step 13: (Optional) Reload All BBX Services" on page 5-21	o				
"Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure" on page 5-23			X	X	X
"Step 15: (Optional) Create MAINVIEW Explorer Host Server Startup Procedure" on page 5-25			o	o	o
"Step 16: (Optional) Migrate CAS Definitions" on page 5-27			o	o	o
"Step 17: (Required) Specify MAINVIEW Alarm Manager Initialization Parameters" on page 5-28			X	X	X
"Step 18: (Required) Allocate MAINVIEW Alarm Manager Journal Logs" on page 5-30			X	X	X
"Step 19: (Required) Allocate MAINVIEW Alarm Manager BBIDIV Data Set" on page 5-32			X	X	X
"Step 20: (Required) Create the MAINVIEW Alarm Manager PAS Startup Procedure" on page 5-33			X	X	X
"Step 21: (Optional) Create MVS PAS Historical Data Sets" on page 5-35			o		

Table 4-2 MAINVIEW Customization Steps (continued)

Customization Checklist	MAINVIEW Product Group				
	All	1	2	3	4
"Step 22: (Required) Create MVS PAS PARMFILE Data Set" on page 5-38			X		
"Step 23: (Required) Create the MVS PAS Startup Procedure" on page 5-39			X		
"Step 24: (Optional) Create BBI-SS PAS Historical Data Sets" on page 5-41				o	
"Step 25: (Optional) Create BBI-SS PAS PARMFILE Data Set" on page 5-44				o	
"Step 26: (Required) Define BBI-SS PAS Suffixes and Target System Parameters" on page 5-46		X		X	
"Step 27: (Optional) Define Configuration Members to BBI-SS PAS" on page 5-51		o			
"Step 28: (Optional) Specify BBI Internal Services Parameters" on page 5-53		o		o	
"Step 29: (Optional) Set BBI-SS PAS Historical Recording Interval" on page 5-56				o	
"Step 30: (Optional) Allocate BBI-SS PAS Journal and Image Logs" on page 5-57		o		o	
"Step 31: (Required) Create a BBI-SS PAS Start Procedure" on page 5-59		X		X	
"Step 32: (Required) Allocate MAINVIEW AutoOPERATOR Data Sets" on page 5-64		X			
"Step 33: (Required) Create Automatic BBI-SS PAS Log Handling Procedures" on page 5-65		X		X	
"Step 34: (Optional) Define Terminal Session (TS) Parameters" on page 5-68		o		o	
"Step 35: (Optional) Define Cross-System Communication" on page 5-72	o				
"Step 36: (Required) Specify Product Initialization within a BBI-SS PAS" on page 5-102		X		X	
"Step 37: (Optional) Enable Automatic Restart Manager Support" on page 5-103		o			
"Step 38: (Required) Specify Product Option Password Keys" on page 5-104		X		X	
"Step 39: (Required) Add CAS ID to BBI-SS PAS" on page 5-109				X	
"Step 40: (Optional) Add Extended Function Data Sets to BBI-SS PAS StartUp Procedure" on page 5-110				o	
"Step 41: (Required) Update TSO Command Processor Table" on page 5-111	X				
"Step 42: (Optional) Specify Product Initialization within the MVS PAS" on page 5-113			o		

Table 4-2 MAINVIEW Customization Steps (continued)

Customization Checklist	MAINVIEW Product Group				
	All	1	2	3	4
(Required) Set up one or more product-specific PASs for products that do not use a BBI-SS or MVS PAS. For implementation instructions, see the customization guides shipped with your products.					X
(Optional) Implement AutoRule (MAINVIEW AutoOPERATOR Only) AutoRule is an optional event management utility. It allows you to create and maintain MAINVIEW AutoOPERATOR Rule Sets, which are built using supplied knowledge bases, input from MVS MPF lists, and other rules you may already have. For implementation instructions, see the <i>MAINVIEW AutoOPERATOR Customization Guide</i> .		o			
(Optional) Implement Security For information about RACF requirements for the terminal session and BBI-SS PAS and securing product resources with other external security managers, see <i>Implementing Security for MAINVIEW Products</i> .	o				
(Optional) Customize Your Product To tailor your product(s) to your site's requirements, see the customization guides shipped with your products.	o				

Chapter 5 Customizing the MAINVIEW Environment

The online environment for MAINVIEW products must be tailored so that the PAS, CAS, terminal session, and target systems can all work together. You can tailor the MAINVIEW environment manually or automatically. BMC Software provides AutoCustomization procedures, described in the *OS/390 and z/OS Installer Guide*, that help you tailor the environment automatically. If you have already tailored your products using AutoCustomization, you do not need to read this chapter.

This section describes the steps for tailoring the MAINVIEW environment manually. If you need any help during AutoCustomization, you can refer to the manual customization step description in this chapter. The step numbers from AutoCustomization might not match the manual customization steps exactly, but the activities of the steps should coincide and be similarly named.

Table 4-2 on page 4-3 is a checklist that summarizes the manual customization steps.

Step 1: (Required) Specify Jobcards and Other Operational Defaults

This section applies to all MAINVIEW products.

In the customization steps for data set allocations, you are instructed to specify the volume serial ID and unit type. In other steps, you are requested to supply the high-level qualifier (*hilevel*) for the product libraries.

Write down the volume ID and the unit type for the product libraries, listing the following information:

Prefix ==> _____ (high-level qualifier used)
Unit-T ==> _____ (generic unit name for temporary data sets)
Unit-P ==> _____ (generic unit name for permanent data sets)
VOLSER ==> _____ (volume ID for permanent data sets)
SYSID ==> _____ (MVS image identifier; not to exceed 8 characters)

The SYSID value is used to build data set names for data sets that cannot be shared across multiple systems. In all cases where *hilevel* is referenced, this value is the *prefix* value for your MAINVIEW product libraries and the SYSID value defined above.

Step 2: (Optional) Review Virtual Storage Requirements

This section applies to all MAINVIEW products.

See the *MAINVIEW Installation Requirements Guide* for estimated storage requirements that apply to your products.

Step 3: (Optional) Generate JCL to Run Installation Verification Program

This section applies to all MAINVIEW products.

BBSAMP Member BBAIVP:

This member contains sample JCL you can use to run program BBAIVP. BBAIVP validates MAINVIEW product installation. The following instructions describe how to use BBAIVP. BBAIVP is a batch job that examines load modules to validate product installation. It reports:

- Any modules that are marked as not executable
- Programs not marked as re-entrant that BBAIVP expected to be marked re-entrant
- Modules that contain unresolved references

BBAIVP does not report weak references (references that are expected to be unresolved).

- The name of the library and the number of modules BBAIVP checked.

To run BBAIVP:

- Step 1** Replace ???????? in BBSAMP member BBAIVP (see the example below) with your high-level qualifier.

The REPORT statement or a SYSPRINT statement, shown in the following example, specifies the data set for the BBAIVP report. The SYSLIB statement specifies the data set for BBAIVP to check.

Figure 5-1 BBAIVP JCL to Validate Product Installation

```
//IVP      EXEC  PGM=BBAIVP
//STEPLIB DD  DISP=SHR,DSN=?????????.BBLOAD
//REPORT   DD  SYSOUT=*
//SYSLIB   DD  DISP=SHR,DSN=?????????.BBLINK
```

- Step 2** Submit BBAIVP for processing.

Step 3 Check your job output for the following condition codes:

- | | |
|----|--|
| 00 | No errors found |
| 08 | One or more unresolved references (see the report for details) |
| 12 | One or more modules not marked as re-entrant that should be (see the report for details) |
| 16 | An operational error
A message was written to the REPORT file or to SYSLOG. |

Note: The product may operate properly even if errors are found; however, you should contact BMC Software Customer Support so they can determine if a problem actually exists.

Step 4: (Optional) Implement GDDM/PGF Support

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

To implement GDDM/PGF support, the GDDM load library should be made available through the system link list; otherwise, ensure that the GDDM load library is available through a STEPLIB or ISPLLIB DD in your logon procedure. With GDDM/PGF enabled, the **GRaph** command provides a high resolution graph of the current view data on terminals that support GDDM graphics. Without this support enabled or on terminals that do not support GDDM graphics, you can view a low resolution graphic of view data.

Step 5: (Optional) Implement Katakana Terminal Support

This section applies to all MAINVIEW products.

To use MAINVIEW products with a katakana terminal:

- The following data sets must be converted to all uppercase characters:
 - BBILIB for installation JCL
 - BBMLIB for messages
 - BBTLIB for tables
- CAPS=Y must be specified for the EXEC statement PARM parameter in the CAS started task procedure (see “Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure” on page 5-23).

BBILIB member @@YZZ001:

This is a utility program that translates lowercase characters to uppercase in these data sets. This utility uses ISPF Table Services to convert the BBTLIB members, executes the Terminal Monitor Program (TMP) in batch mode, and invokes ISPF.

To convert the product data sets and installation JCL to support katakana terminals:

- Step 1** Allocate new *hilevel*.BBILIB, BBMLIB, and BBTLIB data sets to store converted installation JCL, message, and table members. This preserves original distribution members so that non-katakana terminals can use them. Otherwise, messages, views, and panels are displayed in all uppercase on all terminals.

Note: The allocated output data sets must have the same block sizes as the corresponding input data sets.

- Step 2** Modify BBILIB member @@YZZ001 to customize the utility conversion program to your site requirements. Modify the following DD statements to specify input and output data set names for the BBILIB, BBMLIB, and BBTLIB data sets:

DD Name	Description
LCBBLIB	Lowercase input data from BBILIB and BBMLIB
UCBBLIB	Uppercase translated BBILIB and BBMLIB output data
LCBBTLIB	Lowercase input tables from BBTLIB
UCBBTLIB	Uppercase translated BBTLIB output tables

Step 3 Submit the utility program for processing.

Note: You should compress the data sets before submitting the job. If the UCBBLIB and UCBBTLIB DD statements point to the same data sets as the LCBBLIB and LCBRTLIB DD statements, respectively, the original members are overwritten.

Members from the input data sets are read, translated to uppercase, and written to the output data sets. The utility program performs the translations in two stages. At the completion of each stage, the following WTO message is issued when the data translation is successful: CONVERSION SUCCESSFULLY COMPLETED.

Step 4 Check the job output for a condition code of 0 for successful job completion.

Step 6: (Required) Apply Passwords with the Product Authorization Utility

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4 (except for MAINVIEW SRM)

See Table 4-1 on page 4-2 for a list of products in each group.

Note: This step is not required for MAINVIEW for IMS Online or MAINVIEW for DBCTL, if you use BBKEYS authorization to activate the products. For information on BBKEYS authorization, see “Step 38: (Required) Specify Product Option Password Keys” on page 5-104.

This step involves using the BMC Software Product Authorization utility to apply product passwords and update your CPU configuration, if necessary. For a description of the Product Authorization utility and information about applying passwords for your products, refer to the *OS/390 and z/OS Installer Guide*.

Step 7: (Required) Create Site Data Sets for Use with MAINVIEW Products

This section applies to all MAINVIEW products.

Create the following product data sets for your site:

Note: If service needs to be applied, you are alerted when distributed versions change because PTFs that affect any members in these data sets contain the statement:

```
++HOLD SYSTEM REASON(ACTION) CLASS(library-name)
```

You can then save your modifications before replacing the members.

Group 1 Products:

Group1 products are listed in Table 4-1 on page 4-2.

Data Set	Description
UBBPARM	Site-modified parameter members
UBBSAMP	Site-customized sample JCL, procedures, and program members
SBBPROF	(Optional) Site-customized, common profile library shared by all site users

Group 2 and 3 Products:

Group 2 and 3 products are listed in Table 4-1 on page 4-2.

Data Set	Description
UBBPARM	Site-modified parameter members
SBBVDEF	Site-customized view definitions
SBBHDEF	Site-customized help text
SBBPROF	(Optional) Site-customized, common profile library shared by all site users
SBBSDEF	Site-customized screen definitions
UBBSAMP	Site-customized sample JCL, procedures, and program members

Note: Use the same *hilevel* qualifier for these data sets that you used for your target libraries during the install process.

The data set allocations require the attributes shown in the following table:

Data Set	Record Format (RECFM)	Logical Record Length (LRECL)	Block Size (BLKSIZE)	Primary Tracks	Secondary Tracks	Directory Blocks
UBBPARM	FB	80	6160	60	15	80
SBBVDEF	FB	80	6160	30	15	30
SBBHDEF	FB	80	6160	30	15	30
SBBSDEF	FB	80	6160	30	15	30
UBBSAMP	FB	80	6160	30	15	80
SBBPROF	FB	80	6160	30	15	30

BBILIB Member @@YZZ011:

This member contains sample JCL you can use to allocate the data sets described above. To allocate these data sets:

- Step 1** Follow the instructions at the top of @@YZZ011 to modify the JCL to your site requirements.
- Step 2** Submit the job.
- Step 3** Check the job output for a condition code of 0 for successful job completion.

Note: If you use another method to allocate these data sets, such as ISPF's Data Set Utility, do not specify the PACK option. Site product data sets cannot be packed when allocated.

Step 8: (Required for Security) Create Site Security Data Set

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

Default security parameter and resource definition files are distributed in the BBACTDEF library. You cannot customize these defaults. To create your own customized security for MAINVIEW product resources installed at your site, you can use the Plex Manager security views. To use these views, you must define a security library with the BBSECURE DD statement in the CAS and PAS started task procedures. The same library name must be specified in the BBSECURE DD statements in both procedures. This library is used to contain site-customized security resource definitions.

This step provides instructions for creating the security parameter library. For more information about how to set up security and use the MAINVIEW Plex Manager security views, see *Implementing Security for MAINVIEW Products*.

To create a BBSECURE data set:

Step 1 Create a data set with DCB characteristics of RECFM=FB, LRECL=80, and a BLKSIZE of any multiple of 80. If you use ISPF's Data Set Utility, do not specify the PACK option. MAINVIEW product site data sets cannot be packed when allocated.

Note: Use the same *hilevel* qualifier for this data set that you used for your target libraries during the SMP process.

Step 2 Specify the data set name with the DSN parameter of the BBSECURE DD statement in the CAS started task procedure (see “Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure” on page 5-23.).

Step 3 Specify the data set name with the DSN parameter of the BBSECURE DD statement in each PAS procedure (see “Step 23: (Required) Create the MVS PAS Startup Procedure” on page 5-39 and “Step 31: (Required) Create a BBI-SS PAS Start Procedure” on page 5-59).

Step 9: (Required) APF-Authorizing the BMC Software Load Library

This section applies to all MAINVIEW products.

BBILIB Member BAIAPF:

You can use this member as a model to help you temporarily APF-authorize BBLINK programs.

For Temporary Authorization:

Use BBILIB member BAIAPF to temporarily authorize the BBLINK library.
Edit BAIAPF:

- Step 1** Create UBBSAMP (see page 3-6) member BAIAPF.
- Step 2** Copy BBILIB member BAIAPF to the new member.
- Step 3** Change the PROC statement PREFIX parameter from @PREFIX to *prefix*, where *prefix* is the high-level qualifier for the MAINVIEW product libraries.
- Step 4** Change the PROC statement LIB parameter from @BBIAPF to a previously authorized library.
- Step 5** Change the PROC statement VOL parameter from @BBIVOL to the volume where the MAINVIEW product libraries reside.
- Step 6** Submit the job.

For Permanent Authorizatoin:

Use the Authorized Program Facility (APF). Contact a system programmer for assistance in updating either SYS1.PARMLIB member IEAAPF_{xx} or, for SP 5 and above, SYS1.PARMLIB member PROG_{xx}

IEAAPF_{xx}:

where *xx* identifies the APF suffix specified in the IEASYS member used for the last MVS IPL.

An IPL is required before the SYS1.PARMLIB change can take effect.

PROG_{xx}:

where *xx* identifies the member in SYS1.PARMLIB that contains the parameters that define the list of APF-authorized libraries.

In PROG_{xx}, you can specify multiple PROG_{xx} members with the PROG parameter as follows:

```
PROG=(xx,...,xx)
```

where *xx* can be any two alphanumeric characters; for example:

```
PROG= ( 01 , 02 )
```

Libraries are placed in the APF list as follows:

- Libraries specified in the PROG_{xx} member(s).
- If you have an existing IEAAPF_{xx} member and PROG_{xx}, both are processed.

To use only PROG_{xx}, change IEAAPF_{xx} to PROG_{xx} and remove the APF parameters from IEASYS_{xx} and IEASYS00.

- If you have both PROG_{xx} and EXIT_{xx}, PROG_{xx} is processed first then EXIT_{xx}

To use only PROG_{xx}, change EXIT_{xx} to PROG_{xx} and remove the EXIT parameters from IEASYS_{xx} and IEASYS00.

An IPL is required before the SYS1.PARMLIB change can take effect. Or, you can authorize the new libraries with the MVS SET command as follows:

```
T  PROG=xx
```

where *xx* identifies the SYS1.PARMLIB member.

For more information about APF-authorization, see the IBM *MVS/ESA SP V5 Initialization and Tuning Reference* manual.

Step 10: (Optional) Add the BMC Software Load Library to your System Link List

This section applies to all MAINVIEW products.

The BMC Software load library, *hilevel.BBLINK*, can be added to your system link list. BBLINK contains authorized programs.

Notes:

- MVS requires that any data sets added to your system link list be cataloged in the master catalog.
- If you are going to run COMMON STORAGE MONITOR as a subsystem, you *must* perform this step.
- BBLINK should be placed *after* your SAS/C runtime libraries in the system link list.
- If BBLINK is added to the system link list when starting IMS V5.1 and above, the AOI Exits will be dynamically invoked if MAINVIEW AutoOPERATOR is installed whether or not MAINVIEW AutoOPERATOR for IMS is activated.
- If you are not adding BBLINK to your link list, you must specify a STEPLIB DD statement in your MAINVIEW startup procedures (see “Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure” on page 5-23, “Step 23: (Required) Create the MVS PAS Startup Procedure” on page 5-39, and “Step 31: (Required) Create a BBI-SS PAS Start Procedure” on page 5-59).

To add BBLINK to your system link list:

Step 1 Add the following line to your SYS1.PARMLIB(LNKLST $_{xx}$) member:

hilevel.BBLINK,

where *hilevel* is the high-level data set qualifier you used throughout this installation for your BMC Software product data sets.

Step 2 Remove the STEPLIB DD statement for *hilevel.BBLINK* from the MAINVIEW startup procedures, such as the CAS and PAS started task procedures.

Step 3 Update your system link list dynamically.

Step 11: (Required) Create CLIST for Invoking MAINVIEW Products

This section applies to all MAINVIEW products.

If you only have Group 1 products (see Table 4-1 on page 4-2) installed, AutoCustomization uses TSCLIST. To manually customize TSCLIST, you can use BBSAMP member TSCLIST.

BBILIB Member BAIMAINV:

Use this member as a model to help you create a MAINVIEW CLIST to access all MAINVIEW products.

The MAINVIEW CLIST allows you to invoke MAINVIEW products under ISPF. This CLIST allocates all required libraries and connects to them through the ISPF LIBDEF.

To create the MAINVIEW CLIST:

Step 1 Create a member in *hilevel*.UBBPARM called MAINVIEW.

Use of the name MAINVIEW for the CLIST is recommended, but not required. You can use any name you want. The remainder of this procedure uses the name MAINVIEW.

Step 2 Copy the sample CLIST in BBILIB member BAIMAINV into the newly created MAINVIEW member.

Step 3 Edit MAINVIEW CLIST as follows:

- Replace XXXXX in the PREFIX(' XXXXX ') parameter with the *hilevel* qualifier for your target libraries.
- Replace BBBB in the PREFIX(' BBBB ') parameter with the *hilevel*.BBLINK name for your target libraries.
- If you have installed any MAINVIEW products that run in the BBI-SS PAS (see Figure 1-1 on page 1-2), specify the subsystem ID for that BBI-SS with the SS() parameter; otherwise, leave the SS() parameter as is.

- Step 4** If you require optional support for GDDM high resolution graphics, add a TSO ALLOCATE statement for the GDDM symbol library using the ADMSYMBL ddname.

If your level of GDDM supports printing, add a TSO ALLOCATE statement for an optional GDDM PRINT data set using the ADMPRINT ddname. Refer to the GDDM member SYS1.GDDMSAM(ADMQFMT) for more information about how to allocate the GDDM PRINT data set.

- Step 5** To invoke the CLIST to access MAINVIEW products, you can:

- Activate a MAINVIEW Alternate Access VTAM or EXCP session that displays the MAINVIEW Selection Menu (see the *MAINVIEW Alternate Access Implementation and User Guide* for more information)
- Issue a TSO command from any ISPF command line
- Add a menu item to an ISPF menu to execute the MAINVIEW CLIST

To add a menu selection code and description to an ISPF menu:

- 5.A** Add the following statement to the)BODY section of an ISPF menu panel of your choice:

```
% MV +MAINVIEW - Invoke MAINVIEW Products
```

- 5.B** Add the following selection command to the)PROC section of the menu panel for the MV option:

Figure 5-2 Selection Command to Add MAINVIEW to an ISPF Menu

```
)PROC
  &ZSEL = TRANS(TRUNC(&ZCMD, '.'))
  .
  MV, 'CMD(EX ''hilevel.UBBPARM(MAINVIEW)'' )'
  .
  .
  X, 'EXIT'
  *, '?' )
```

Step 12: (Optional) Add TSF Authorization for Products to Run Under ISPF

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

To run your products in both windows of an ISPF split-screen session, it is necessary to

- add TSF authorization
- add BBLINK to either STEPLIB or LINKLIST

Otherwise, split screen access is only available when the products are invoked from native TSO.

TSF authorization is required for the MAINVIEW **TRANSfer** command that allows users to switch to product services that operate in the ISPF panel environment.

To add TSF authorization:

Step 1 Add PMGLAUTH to the AUTHTSF list in SYS1.PARMLIB(IKJTSO_{xx}).

Step 2 To ensure addressability to the libraries, do one of the following:

- 2.A** Concatenate BBLINK to the STEPLIB in the ISPF logon procedures to be used for your BBI-based products. This requires that all libraries within the STEPLIB concatenation be APF-authorized.
- 2.B** Add BBLINK to your system link list.

Step 13: (Optional) Reload All BBX Services

This section applies only to the following products:

- CMF MONITOR
- InTune
- MAINVIEW AutoOPERATOR for MVS
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for OS/390
- MAINVIEW SYSPROG Services
- MAINVIEW VistaPoint

BBXS is a set of common service routines used by all of the above products.

BBXS service routines return data in response to specific requests from these products. BBXS is initialized the first time a product requests a service or when the COMMON STORAGE MONITOR component of MAINVIEW for OS/390 is configured to initialize at IPL.

Warning! Do not add:

```
COM= ' S BBXSINIT '
```

to the SYS1.PARMLIB member COMMNDxx unless you are running under VM.

For more information about BBXS, see the *MAINVIEW Administration Guide*.

BBILIB Member @@YZZ081:

You can use this member to re-initialize BMC Software Subsystem Services (BBXS) when a more current version of BBXS is installed.

To re-initialize BBXS:

Step 1 Follow the instructions at the top of BBILIB member @@YZZ081 to modify the JCL to your site requirements.

Note: If you are running MVS under VM, you must add the IOCDS parameter to the EXEC statement in member @@YZZ081 before submitting this job. This parameter identifies the resident IOCDS ID that BBXS uses during initialization.

The IOCDS parameter may change frequently in some environments. For the JCL in this step to be accurate, it must be modified to reflect any IOCDS changes. Contact the person responsible for IOCDS generations at your site to obtain the current IOCDS ID. For more information about initializing BBXS in MVS under VM, see the *MAINVIEW Administration Guide*.

Step 2 Submit the job.

Step 3 Check the job output for a condition code of 0 for successful job completion.

If the condition code is 4 or 12, review the job output messages. For an explanation of BBXS messages, see the MAINVIEW online message system. A condition code of 0 with message BBX600I indicates successful completion.

Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

BBILIB Member @@YZZ021

You can use this member as a model to help you set up your CAS started task procedure.

To execute the CAS:

- Initiate it as a started task (STC) through an MVS IPL procedure
- Execute only one CAS per MVS image
- Execute the CAS with a dispatching priority of 255

The CAS contains functions accessed through MVS cross-memory services and should not be run as a batch job. If the CAS runs as a batch job:

- It operates correctly, but the MVS job scheduler issues messages IEF353A and IEF355A upon CAS termination
- The JES initiator running the CAS terminates

To create the CAS started task procedure:

Step 1 Create a CAS startup member in a system procedure library.

Note: AutoCustomization creates a member named BBMCAS by default.

Step 2 Copy BBILIB member @@YZZ021 to the member you created in Step 1.

Step 3 Follow the instructions at the top of this member to modify the procedure to your site requirements.

For a description of each statement in this procedure, see Appendix A “Library Member for CAS” on page A-1.

Step 4 Optionally, define the CAS started task procedure to your IPL procedure.

You may want to modify the `COMMNDxx` member in `SYS1.PARMLIB` to add a `START` command for the CAS procedure. The `COMMNDxx` member contains MVS commands that are issued by the master scheduler upon system initialization. The `START` commands are issued in the order they appear in `COMMNDxx`

The format of the command is:

```
COM= 'S procname ,SUB=MSTR '
```

where *procname* is the name of the member you created in Step 1 on page 5-23; for example:

```
COM= 'S BBMCAS ,SUB=MSTR '
```

Required SSID Parameter:

When you create your CAS started task procedure, you must specify a subsystem ID for the CAS with the `PROC` statement `SSID` parameter. This value identifies the CAS to MVS and is used by all other applicable PASs.

The CAS subsystem ID is used:

- In the Session Control Parameters dialog when the TS starts
- In the startup procedure of the PAS that connects to this CAS

Note: The CAS and PAS started task procedures and the Session Control parameters dialog must all define the same subsystem ID.

Optional CAPS Parameter:

If you use a katakana terminal (see “Step 5: (Optional) Implement Katakana Terminal Support” on page 5-7), you must specify `CAPS=Y`, with the `EXEC` statement `PARM` parameter in your CAS started task procedure; for example:

```
//CAS EXEC PGM=BBM9ZA00,  
// PARM=( 'SSID=&SSID ,XDM=&XDM ,COLD=&COLD ,DUMP=&DUMP ,EMM=&EMM' ,  
// 'IVP=&IVP ,CAPS=Y' ) ,  
// TIME=1439 ,  
// REGION=4096K
```

Step 15: (Optional) Create MAINVIEW Explorer Host Server Startup Procedure

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

MAINVIEW Explorer is a client/server application that lets you access MAINVIEW products from a Web browser. This step helps you customize the MAINVIEW Explorer host server to your site's needs.

BBSAMP Member BBMXPJCL:

Use this member as a model for the MAINVIEW Explorer host server startup procedure.

Step 1 Copy BBSAMP member BBMXPJCL to a procedure library.

You can change the member name if desired. The member name is the name you will specify on the MVS START command when you start the host server.

Step 2 Change `SSID=?SSID` to `SSID=cas`, where *cas* is the subsystem ID of the CAS to which MAINVIEW Explorer is to connect.

Step 3 Change all other occurrences of ? to *hilevel.*, where *hilevel* is the high-level qualifier used for BMC Software product data sets.

Step 4 If you changed the member name, update the name on the PROC statement in the JCL to match the new member name.

Step 5 Save the member.

Note: BBMXPJCL contains the procedure variable `PORT=3940`, which is the default host server Port ID. If you change the Port ID in BBMXPJCL, you must specify the new value when you start MAINVIEW Explorer.

BBSAMP Member BBTTCP00:

Use this member if you want to connect to a secondary TCP/IP stack or debug TCP/IP related problems.

Copy BBSAMP member BBTTCPxx to UBBPARM and modify it with the desired parameters. Then rename the modified member to BBTTCP00 .

See “BBPARM Member BBTTCP00 Parameters” on page C-9 for a description of the parameters.

Step 16: (Optional) Migrate CAS Definitions

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

If a new version of MAINVIEW Infrastructure is installed separately from any other MAINVIEW product or you are migrating from an earlier version of MAINVIEW Infrastructure, you might want to migrate your existing CAS definitions. CAS definitions are stored in member BBMTYB00 of the *hilevel.UBBPARM* data set. Copying this member forward will enable the new CAS that you are setting up to know about all of your existing CASs.

To migrate your existing CAS definitions, copy member BBMTYB00 from your existing *hilevel.UBBPARM* data set to the new *hilevel.UBBPARM* data set that was created for MAINVIEW Infrastructure.

Step 17: (Required) Specify MAINVIEW Alarm Manager Initialization Parameters

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to enable MAINVIEW Alarm Manager for initialization within its own PAS.

Note: MAINVIEW Alarm Manager can not reside in the same BBI-SS PAS as any other MAINVIEW product.

BBPARM Member BBISSP00:

MAINVIEW Alarm Manager requires its own BBISSP00 member. This member can not be shared by any other BBI-SS PAS. Other considerations include the following:

- The BBISSP00 member for MAINVIEW Alarm Manager resides in a separate parameter library with a low-level qualifier of MVAPARM.
- You should not include the UBBPARM data set in the BBPARM concatenation of the MAINVIEW Alarm Manager PAS JCL.

To update BBISSP00 for MAINVIEW Alarm Manager:

Step 1 Specify the following PRODUCT parameter:

PRODUCT=MVA

Note: PRODUCT=MVA must be the *only* PRODUCT parameter in the BBISSP00 member for MAINVIEW Alarm Manager. If any other PRODUCT parameters are found, MAINVIEW Alarm Manager will not initialize.

Step 2 Specify the ALRTRCVE parameter as follows:

ALRTRCVE=YES | NO

Specify YES to indicate that this MAINVIEW Alarm Manager PAS is eligible to receive alerts from other BMC Software products. The default is NO.

Step 3 Specify the CASID parameter as follows:

CASID=*ssid*

where *ssid* is the 4-character subsystem ID of the CAS to which the MAINVIEW Alarm Manager PAS is to connect.

BBPARM Member MVAPRM00:

Specify the ALRMSFX parameter as follows:

ALRMSFX=*xx*

where *xx* is the suffix of the MAINVIEW Alarm Manager member to use. To identify more than one member, specify the parameter like this:

ALRMSFX=(00 ,*xx*,...)

Step 18: (Required) Allocate MAINVIEW Alarm Manager Journal Logs

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to allocate journal log data sets that MAINVIEW Alarm Manager uses to record:

- commands and responses issued by the Alert Management component
- product alerts, warnings, and messages

BBSAMP Member SSLOG:

This member contains sample JCL to allocate MAINVIEW Alarm Manager journal log data sets.

Notes:

- A set of journal log data sets must be allocated for every MAINVIEW Alarm Manager PAS.
- Do not alter the log attributes because the defaults are required and are reset automatically if you change them.
- If you are using products on multiple systems and want to use the same JCL procedure for all of them, use the SSID in the data set name of the logs. All the subsystems can then use the same PROC.

To allocate the MAINVIEW Alarm Manager journal log data sets:

Step 1 Globally change the following:

- Change the PREFIX parameter to the prefix used to load the product libraries to DASD.
- Set the SS parameter value to the subsystem ID of the MAINVIEW Alarm Manager PAS.
- Set the DV parameter value to the volume serial of the allocation volume for the journal data sets.
- Set the DU parameter value to the allocation volume unit type.

Step 2 Submit the newly modified job.

Step 19: (Required) Allocate MAINVIEW Alarm Manager BBIDIV Data Set

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to allocate the data-in-virtual data set (BBIDIV) required for MAINVIEW Alarm Manager. The BBIDIV data set is used to store binary large objects and data (such as non-volatile alerts) across restarts.

BBSAMP Member DIVDEF:

Use this member to allocate a BBIDIV data set, which is required by the Alert Management component of MAINVIEW Alarm Manager.

Notes:

- Each MAINVIEW Alarm Manager PAS must have its own copy of this data set; it cannot be shared.
- The MAINVIEW Alarm Manager PAS must be down when this data set is allocated.

To allocate the MAINVIEW Alarm Manager BBIDIV data set, make the following changes to the procedure header:

- Step 1** Change *HILEVEL* to the high-level qualifier for MAINVIEW product data sets.
- Step 2** Change *SSID* to the 4-character subsystem name.
- Step 3** Change *VOL* to the volume serial number of the DASD volume where the linear data set will be allocated.

Step 20: (Required) Create the MAINVIEW Alarm Manager PAS Startup Procedure

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3
- Group 4

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to set up a started task procedure for the MAINVIEW Alarm Manager PAS.

Notes:

- BMC Software recommends that you run only one MAINVIEW Alarm Manager PAS per MVS image.
- The recommended dispatching priority of the MAINVIEW Alarm Manager PAS is 255.

BBSAMP Member BBHPAS:

Use this member to help you set up your MAINVIEW Alarm Manager PAS started task procedure.

Step 1 Copy BBSAMP member BBHPAS to SYS1.PROCLIB or another system procedure library.

Step 2 Follow the instructions at the top of the member to modify the procedure for your site requirements.

Note: See Table 4-1 on page D-1 for an explanation of the DD statements; see Table 4-2 on page D-2 for an explanation of the PROC symbolic parameters.

Step 3 Optionally, define the MAINVIEW Alarm Manager PAS to your IPL procedure, so the PAS starts at system initialization.

BMC Software recommends that you modify SYS1.PARMLIB member COMMND:xx to define the MAINVIEW Alarm Manager PAS to initialize at IPL. Add the following START command to member COMMND:xx to define the PAS to your IPL procedure:

```
COM='S procname'
```

where *procname* is the name of the MAINVIEW Alarm Manager PAS started task procedure.

Note: These commands are issued in the order in which they appear in *COMMNDxx*. The PAS START command must occur after the CAS START command. See “Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure” on page 5-23 for more information about defining the CAS procedure.

If you do not define the MAINVIEW Alarm Manager PAS to start at IPL, see the *MAINVIEW Administration Guide* for instructions on manually issuing a START command for the PAS.

Required Security Updates:

MAINVIEW Alarm Manager uses the AOAnywhere facility to display product alerts. AOAnywhere, which is packaged with all MAINVIEW products, enables certain automation functions to be shared by those products.

To use AOAnywhere, the MAINVIEW Alarm Manager PAS must be associated with a user ID that has READ access to the following resources:

```
prefix.ssid.BBI.target.ACCESS  
prefix.ssid.AAO.target.AOEXEC.ALRTREAD  
prefix.ssid.AAO.target.AOEXEC.ALRTUPD  
prefix.ssid.AAO.target.EXEC
```

For information about securing these resources, see the table called “Resources Used by Multiple Products” in the Basic Security section of *Implementing Security for MAINVIEW Products*.

In addition, if you run MAINVIEW AutoOPERATOR and you plan to launch EXECs from alarm definitions, MAINVIEW Alarm Manager must have the authority to schedule those EXECs. For information about granting access to specific EXECs, see the discussion of MAINVIEW AutoOPERATOR EXECs in *Implementing Security for MAINVIEW Products*.

Step 21: (Optional) Create MVS PAS Historical Data Sets

This section applies only to Group 2 products. See Table 4-1 on page 4-2 for a list of products in this group.

All products that run in the same MVS PAS share the same historical data sets. If you have already defined historical data sets for an MVS PAS product, you do not need to do this step.

BBILIB Members @ @YZZ051 and @ @YZZ052:

You can use these members as models to help you allocate and format MVS PAS historical data sets.

Historical data sets are VSAM files that store data at the end of a recording interval. You can display historical data in Group 2 product views only if historical data sets are allocated.

Note: If you do not want to use historical data, delete the HISTDSnn DD statements from the PAS started task procedure described in “Step 23: (Required) Create the MVS PAS Startup Procedure” on page 5-39.

To allocate and format the MVS PAS historical data sets:

Step 1 Determine how many VSAM files you need to define and how large the files should be.

Default customization provides for three historical data sets, although you can allocate a maximum of 100 data sets. When one data set is full, recording starts with the next data set until all data sets are full. Recording then begins again with the first allocated data set and overwrites existing data previously recorded there.

The appropriate number of historical data sets for your site depends on several factors, including system load, system configuration and the specific MAINVIEW products that are sharing the MVS PAS. BMC Software suggests that you follow this procedure to determine the historical data set space requirements for your site:

1.A Decide how many historical data sets you want to use.

There is no rule about how many historical data sets a site should have. At some sites the number is based on the number of times the system workload characteristics change during the day. That number is often three, corresponding to shift changes. However, you may decide a different number is appropriate for your site.

1.B Determine the size of the historical data sets.

Initially, BMC Software recommends that you begin with three 30-cylinder data sets (on a 3390). Once you complete all product customizations and begin to run the PAS, allow the PAS to run for approximately 24 hours. During that time, make note of when the PAS receives messages indicating that the historical data set is full and is being switched. By doing so you can determine how much space is typically required to hold historical data over a given amount of time. You can then adjust the sizes of the historical data sets if you want them to hold data for a longer or shorter period of time.

Note: You can add historical data sets dynamically from the DSLIST view, as described in the *MAINVIEW Administration Guide*.

Step 2 Follow the instructions at the top of BBILIB member @@YZZ051 to modify the JCL according to your site's needs.

Step 3 Submit the job to allocate and format the historical data sets. The data sets are allocated and then formatted using an initial record format shipped in BBILIB member @@YZZ052.

The data sets are allocated using a default data set name of *hilevel.SMFID.HISTDSnn*, where:

hilevel Is the high-level qualifier for your BMC Software product libraries.

SMFID Is the SMF identifier of the system for which the data will be collected.

nn Is a unique value from 00 to 99. Each historical data set must have a unique name.

Step 4 Check the job output for a condition code of 0 for successful job completion.

If the condition code is not 0, review the job output messages.

Step 5 Ensure that there is a `//HISTDSnn DD` statement defined in the PAS started task procedure (see “Step 23: (Required) Create the MVS PAS Startup Procedure” on page 5-39) for each historical data set that you allocate.

Step 22: (Required) Create MVS PAS PARMFILE Data Set

This section applies only to Group 2 products. See Table 4-1 on page 4-2 for a list of products in this group.

BBILIB Member @@YZZ061:

You can use this member as a model to help you create a VSAM data set required for historical data processing and workload configuration definitions.

A PARMFILE data set is a VSAM data set that stores the names and current status of historical data sets. When historical data sets are listed in PARMFILE, you can dynamically enable them with commands from the DSLIST view.

A separate PARMFILE data set is required for each PAS. To allocate a PARMFILE data set:

- Step 1** Follow the instructions at the top of BBILIB member @@YZZ061 to modify the JCL to your site's requirements.
 - Step 2** Submit the job.
 - Step 3** Check the job output for a condition code of 0 for successful job completion.
- If the condition code is not 0, review the job output messages.

Step 23: (Required) Create the MVS PAS Startup Procedure

This section applies only to Group 2 products. See Table 4-1 on page 4-2 for a list of products in this group.

BBILIB Member BBMPAS:

Use this member as a model to help you set up your MVS PAS started task procedure.

To execute the PAS:

- Initiate it as a started task (STC) through an MVS IPL procedure
- Execute only one MVS PAS per MVS image
- Execute the MVS PAS with a dispatching priority of 255

Required SSID Parameter:

Specify the same subsystem ID with the PROC statement SSID parameter as you did in the CAS started task procedure. This is used by the PAS during initialization to identify its CAS connection.

To create the MVS PAS started task procedure:

Step 1 Create an MVS PAS startup member in SYS1.PROCLIB or another system procedure library.

Note: AutoCustomization creates a member named MVSPAS by default.

Step 2 Copy BBILIB member BBMPAS to the member you created in Step 1 on page 5-23.

Step 3 Follow the instructions at the top of this member to modify the procedure to your site requirements.

For a description of each statement in this procedure, see Appendix B“Library Member for MVS PAS” on page B-1.

Step 4 Optionally define the MVS PAS started task procedure to your IPL procedure.

Modify the `COMMNDxx` member in `SYS1.PARMLIB` to add a `START` command for the PAS procedure. The `COMMNDxx` member contains MVS commands that are issued by the master scheduler upon system initialization. The `START` commands are issued in the order they appear in `COMMNDxx`

The MVS PAS `START` command must occur after the `CAS START` command (see “Step 14: (Required) Create the CAS (Coordinating Address Space) Startup Procedure” on page 5-23).

If you do not define the MVS PAS to start at IPL, you must manually enter a `START` command for the PAS (see the *MAINVIEW Administration Guide*).

The format of the command is:

```
COM= 'S procname '
```

where *procname* is the name of the member you created in Step 1 on page 5-23; for example:

```
COM= 'S MVSPAS '
```

Step 24: (Optional) Create BBI-SS PAS Historical Data Sets

This section applies only to Group 3 products. See Table 4-1 on page 4-2 for a list of products in each group.

All products that run in the same BBI-SS PAS share the same historical data sets. If you have already defined historical data sets for a BBI-SS PAS product, you do not need to do this step.

BBILIB Members @ @YZZ051 and @ @YZZ052:

You can use these members as models to help you allocate and format BBI-SS PAS historical data sets.

Historical data sets are VSAM files that store data at the end of a recording interval. You can display historical data in Group 3 product views only if historical data sets are allocated.

You must allocate a separate set of historical data sets for each BBI-SS PAS.

You cannot store BBI-SS PAS data in historical data sets allocated to the MVS PAS (see “Step 21: (Optional) Create MVS PAS Historical Data Sets” on page 5-35).

To allocate and format the BBI-SS PAS historical data sets:

- Step 1** Determine how many VSAM files you need to define and how large the files should be.

Default customization provides for three historical data sets, although you can allocate a maximum of 100 data sets. When one data set is full, recording starts with the next data set until all data sets are full. Recording then begins again with the first allocated data set and overwrites existing data previously recorded there.

The appropriate number of historical data sets for your site depends on several factors, including system load, system configuration and the specific MAINVIEW products that are sharing the MVS PAS. BMC Software suggests that you follow this procedure to determine the historical data set space requirements for your site:

1.A Decide how many historical data sets you want to use.

There is no rule about how many historical data sets a site should have. At some sites the number is based on the number of times the system workload characteristics change during the day. That number is often three, corresponding to shift changes. However, you may decide a different number is appropriate for your site.

1.B Determine the size of the historical data sets.

Initially, BMC Software recommends that you begin with three 30-cylinder data sets (on a 3390). Once you complete all product customizations and begin to run the PAS, allow the PAS to run for approximately 24 hours. During that time, make note of when the PAS receives messages indicating that the historical data set is full and is being switched. By doing so you can determine how much space is typically required to hold historical data over a given amount of time. You can then adjust the sizes of the historical data sets if you want them to hold data for a longer or shorter period of time.

Note: You can add historical data sets dynamically from the DSLIST view, as described in the *MAINVIEW Administration Guide*.

Step 2 Copy member @@YZZ051 from BBILIB to a UBBSAMP data set.

@@YZZ051 contains sample JCL to allocate three VSAM data sets with IDCAMS DEFINE CLUSTER statements.

Step 3 Edit the @@YZZ051 member copied to your UBBSAMP data set.

Change all instances of ?BBCHILV to the high-level qualifier name chosen for your historical data sets. Each historical data set must have a unique name.

Data sets are allocated using a data set name of *hilevel.ssid.HISTDSnn*, where:

<i>hilevel</i>	High-level qualifier selected for this library
<i>ssid</i>	The subsystem ID of the BBI-SS PAS
<i>nn</i>	Two-digit value from 00 to 99

Change all instances of ?BBASMFID to the subsystem ID of the BBI-SS PAS. You must use this ID if your site runs more than a single PAS per MVS image.

Change all instances of ?BBPVOL to the volume name where the data sets will be allocated.

Change all instances of ?CYL0, ?CYL1, and ?CYL2 to the number of cylinders allocated for corresponding primary, secondary, and tertiary data sets.

The number of cylinders you need for historical data sets depends upon the number of targets and monitors defined by your site. You should use fewer large data sets rather than more smaller data sets. If you are using three historical data sets, set the initial allocation to 50 cylinders each. Adjust the data set cylinder value accordingly after you have some experience with the storage you actually need to save historical records.

Step 4 Submit the @@YZZ051 job to allocate and format the historical data sets.

@@YZZ051 allocates the data sets then formats them according to an initial record format shipped in BBILIB member @@YZZ052.

Check the job output for a condition code of 0 for successful job completion.

If the condition code is not 0, review the job output messages.

Step 25: (Optional) Create BBI-SS PAS PARMFILE Data Set

This section applies only to Group 3 products. See Table 4-1 on page 4-2 for a list of products in each group.

BBILIB Members @@YZZ061 and @@YZZ062:

You can use these members as models to help you allocate and format a PARMFILE data set for the group 3 products shown in Table 4-1 on page 4-2.

A PARMFILE data set is a VSAM data set that stores the names and current status of historical data sets. When historical data sets are listed in PARMFILE, you can dynamically enable them with commands from the DSLIST view.

A separate PARMFILE data set is required for each PAS. To allocate a PARMFILE data set:

Step 1 Copy member @@YZZ061 from BBILIB to a UBBSAMP data set.

@@YZZ061 contains sample JCL to allocate a VSAM data set with IDCAMS DEFINE CLUSTER statements.

Step 2 Edit the @@YZZ061 member copied to your UBBSAMP data set.

- Change all instances of ?BAVSHLQ to the high-level qualifier name chosen for your PARMFILE data set.
- Change all instances of ?BBCHILV to the high-level qualifier chosen for your installation libraries.
- Change all instances of ?BAVIMID to the subsystem ID of the BBI-SS PAS.
- Change all instances of ?BBVOL to the volume name where the data sets will be allocated.
- Change all instances of ?BBPUNIT to the appropriate unit name of the volume storing the PARMFILE data set.

Step 3 Submit the @@YZZ061 job to allocate and format the PARMFILE data set.

@@YZZ061 allocates the data set then formats it according to an initial record format shipped in BBILIB member @@YZZ062.

Check the job output for a condition code of 0 for successful job completion.

If the condition code is not 0, review the job output messages.

Step 26: (Required) Define BBI-SS PAS Suffixes and Target System Parameters

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to define all eligible target systems and associate them with the subsystem IDs of their BBI-SS PASs. For this step, you need to

- decide which target you want to associate with which BBI-SS PAS
- specify the targets and BBI-SS PASs in BBPARM member BBIJNT00

Alternatively, for MAINVIEW for CICS, an administrative dialog can be used to define and add targets (see “Managing Targets and Target Contexts” in the *MAINVIEW Administration Guide*). Target definitions 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 active on other PASs from the CAS. It appends this information to a list of targets built during startup from BBPARM member BBIJNT00.

Note: Existing targets defined in BBIJNT00 cannot be changed with the administrative dialog. If this dialog is used to change targets already defined in BBIJNT00, the change is ignored.

Step 1 Write down the name of your subsystem and its associated target, using the sample form shown in the following figure. Use this form to list your subsystem ID and target job names.

Subsystem Name	TARGET	TYPE	RELEASE (DB2, IMS)	IMSTYPE (DBDC, DBCTL)	DESCRIPTION
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____

There is no limit to the number of subsystems that can be defined. You will use this information in “To specify a target system for this BBI-SS PAS, enter:” on page 5-48.

Step 2 Define a 1- to 4-character subsystem ID to dynamically establish a formal MVS subsystem. The name

- must be unique from any other subsystem name and cannot be in the MVS subsystem name table (IEFSSNxx)
- cannot be DB2, IMS, CICS, or JES, which are reserved by MVS
- must not be the same as any other BBI-SS PAS or CAS currently in use

Note: If the subsystem ID you specify is in IEFSSNxx, message SS0974E is issued and initialization terminates.

Step 3 Define a 1- to 8-character job name for each target to be managed by a BBI-SS PAS.

If the target is a DB2 system, specify the 1- to 4-character subsystem name of that DB2.

If the target is a queue manager, specify the 1- to 4-character subsystem name of that queue manager.

Each BBI-SS PAS can manage the following targets on a single MVS image:

- 32 CICS regions
- one IMS control or DBCTL region
- 15 DB2 regions at the same release level
- the MVS system where the BBI-SS PAS runs
- multiple queue managers

- Step 4** Define the type of target, which can be CICS, DB2, IMS, MVS, or QMGR.
- Step 5** For IMS or DB2 targets, identify the target system release.
- Step 6** (Optional) Provide a 1- to 22-character description of the target system.
- Step 7** For IMS targets, identify the type of target as DBDC or DBCTL.
- Step 8** Use the information you wrote down from the steps above to modify the parameters in your copy of BBPARM member BBIJNT00, as described below. These parameters specify a target/BBI-SS PAS pair and other systems with which this BBI-SS PAS must connect. You can specify multiple TARGET statements for different target systems monitored from the same BBI-SS PAS.

BBPARM Member BBIJNT00:

Use this member to define your targets for an associated BBI-SS PAS. The following conventions must be followed when editing member BBIJNT00:

- all statements must be contained within columns 1 – 71
- statements can be continued across multiple lines
- if a statement is continued, the TARGET parameter must appear on the first line
- comments, which begin with an asterisk (*) in column 1, can be inserted anywhere, even between continued statements
- the TARGET parameter and comments must begin in column 1; continuations do not have to begin in column 1

To specify a target system for this BBI-SS PAS, enter:

```
TARGET=tgtname , TYPE=type ,  
IMSTYPE=systype ,  
SUBSYS=subsys , RELEASE=0vr0  
[ , DESC=desc ][ , IMSID=ssid ][ , ALIAS=name ][ , ATTACH=IMM | DEFER]
```

where

TARGET Specifies a 1- to 8-character name of a target system to be monitored by a BBI-SS PAS. *tgtname* can be specified as follows:

For MAINVIEW AutoOPERATOR, specify the name of a target CICS, IMS, or MVS, such as TARGET=MVSPROD .

For MAINVIEW for WebSphere MQ, specify the queue manager subsystem ID (1 to 4 characters), such as TARGET=CSQ1. It can be any name used to relate to a non-MVS queue manager name.

For MAINVIEW for CICS, specify the job or step name of a target CICS, such as TARGET=CICSA1 .

For MAINVIEW for IMS Online, specify the name of a target IMS, such as TARGET=IMS71X .

For MAINVIEW for DBCTL, specify the job name or IMS ID of the DBCTL region, such as TARGET=IMSDBCP .

For MAINVIEW for DB2, specify a DB2 subsystem ID (1 to 4 characters), such as TARGET=DB2P .

TYPE Is the type of target. For the following products, the TYPE parameter can be used as shown.

For MAINVIEW AutoOPERATOR, specify TYPE=CICS | IMS | MVS for a CICS, IMS, or MVS target.

For MAINVIEW for WebSphere MQ, specify TYPE=QMGR for a queue manager target.

For MAINVIEW for CICS, specify TYPE=CICS for a CICS target.

For MAINVIEW for IMS Online, specify TYPE=IMS for an IMS target.

For MAINVIEW for DBCTL, specify TYPE=IMS for an IMS target.

For MAINVIEW for DB2, specify TYPE=DB2 for a DB2 target.

IMSTYPE Where *systype* identifies the type of IMS target. For MAINVIEW for IMS Online, specify IMSTYPE=DBDC | DBCTL for an IMS DB/DC system (default) or a DBCTL system.

The IMSTYPE parameter is valid only when TYPE=IMS is specified.

SUBSYS Where *subsys* represents a subsystem ID assigned to a BBI-SS PAS (maximum of four characters). If only one BBI-SS PAS is created, use the value SSA1.

RELEASE For IMS, DB2 and DBCTL targets, specify the 4-digit release number of the target.

For example, `RELEASE=0610` specified with `TYPE=DB2` indicates DB2 release 6.1.

DESC (Optional) For MAINVIEW FOCAL POINT, specify a 1- to 22-character description of the target system. If the description contains any embedded commas or blanks, it must be enclosed in single quotes, like this:

`'xxx ... xxx'`

IMSID (Optional) For IMS targets, specify a unique 1- to 4-character subsystem ID of the IMS control region that you want to monitor.

Note: If you specify `IMSID`, you cannot specify `ALIAS`.

ALIAS For the following products, the `ALIAS` parameter can be used as follows:

MAINVIEW AutoOPERATOR for IMS, MAINVIEW for DB2, or
MAINVIEW for IMS Online

Specify a 1- to 8-character alias name for the target subsystem. If you have two targets with the same `tgtname` controlled by different BBI-SS PASs, you must define an alias name, such as `ALIAS=PRODDDB2`.

Note: If you specify `ALIAS` for an IMS target, you cannot specify `IMSID`.

MAINVIEW for CICS

Specify the 1- to 8-character `APPLID` specific to the CICS region.

ATTACH For Group 3 products (see Table 4-1 on page 4-2), the `ATTACH` parameter requests immediate or deferred target recognition when the BBI-SS PAS starts.

`DEFER` (the default) causes the service point to be created once the target is active. Data is not supplied in response to queries until the target becomes active or a monitor is started that collects data from that region. In most cases, the `ATTACH=DEFER` should be used.

`IMM` causes the service point to become active when the BBI-SS PAS initializes. It remains active regardless of the target state, which is useful for viewing historical data from that target when it is no longer active. The target's status is shown as active in the Plex Manager views. The PAS can collect data from the target, but that target itself is not active.

Step 27: (Optional) Define Configuration Members to BBI-SS PAS

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

BBPARM Member BBIBBCFG:

Use this member as a model for the CFG*ssidA* member. CCF*ssidA* defines the member name suffixes of the BBPARM configurations members. These members control processing such as:

- How MAINVIEW AutoOPERATOR ALERTs use storage
- How the Automation Reporter application operates
- How MAINVIEW AutoOPERATOR EXECs operate on your system
- How the MAINVIEW AutoOPERATOR TapeSHARE application operates
- How the BBI-SS PAS operates

These members are described in “BBPARM Member BBIBBCFG Parameters” on page C-2.

Specifying different member name suffixes allows multiple BBI-SS PASs to process different configuration members from the same BBPARM library. The CFG*ssidA* member is read during BBI-SS PAS startup to determine the suffixes of configuration members to be used by that PAS.

To define the configuration members to the BBI-SS PAS:

- Step 1** Create a member in your UBBPARM data set named CFG*ssidA*, where *ssid* is the BBI-SS PAS ID.
- Step 2** Copy the distributed BBPARM member BBIBBCFG into the member you just created.

In this member, you can specify:

```
AAOALS=xx
AAOARP=xx
AAOEXP=xx
AAOGME=xx
AAOMQL=xx
```

AAOPRM=*xx*
AAOTSP=*xx*
BBISSP=*xx*

where *xx* is a 2-character member suffix. See page C-2 for an explanation of these members.

- Step 3** Add a BBCFG DD statement that refers to *CFGssidA* to the BBI-SS PAS startup procedure, as described in “Step 31: (Required) Create a BBI-SS PAS Start Procedure” on page 5-59.

Step 28: (Optional) Specify BBI Internal Services Parameters

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3 (except for MAINVIEW for WebSphere MQ)

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to define parameters to initialize timer services automatically at BBI-SS startup and specify a trace directory data set.

BBPARM Member BBIISP00:

Use this member to help you set up the timer services and the trace directory.

- Step 1** Specify the name of a BBPARM member containing a group of requests for timer-driven services to be started at BBI-SS PAS startup.

```
AUTOID=userid
TARGET=tgtname , BLK=blkname [ , USRID=userid]
```

where:

AUTOID	Specifies the logon ID of a user to be associated with automatically started timer requests. AUTOID is the default if USERID is not specified. AUTOID must be on a separate line from the TARGET statement.
TARGET	Specifies the target system to receive the requests for BLK. It can be either the target system ID or the subsystem ID. TARGET must be on the same line as BLK and can be repeated.
BLK	Specifies a 1- to 8-character name of a BBPARM member that defines a collection of timer-driven SET requests for product services that can be initiated at the same time. BLK must be on the same line as TARGET.

The members are:

For MAINVIEW for CICS

- CMRBLK1 Contains a group of background service requests that are used when MAINVIEW for CICS data collection is not active.
- CMRBLK2 Contains a group of background service requests that are used when MAINVIEW for CICS data collection is active.
- CMRBLK3 Contains a group of background service requests that are used when MAINVIEW for CICS data collection is active and MAINVIEW FOCAL POINT is installed.

For MAINVIEW for DB2, MAINVIEW for DBCTL, and MAINVIEW for IMS Online

The following BBPARM members can be used as a base from which to build a list of monitors to be started automatically. These members define suggested warning thresholds; some show only your site's activity.

- BLKDMRW (MAINVIEW for DB2) Can be used to create a starter set of monitors and traces to provide recent DB2 history.
- BLKIMFW (MAINVIEW for IMS Online) Can be used to create a starter set of monitors for IMS.
- BLKDBTW (MAINVIEW for DBCTL) Can be used to create a starter set of monitors for a DBCTL system.

USRID

Specifies the logon ID of a user to be associated with the timer requests in the member specified with the BLK parameter. This user has the authority to modify or purge these requests.

The value for *userid* must match a BBPARM user authorization member (see BBPARM member USERID).

AUTOID is required and is written on a separate line from TARGET and BLK; for example:

```
AUTOID=ADMIN
TARGET=DB2A , BLK=BLKDB2A
TARGET=DB2B , BLK=BLKDB2B
```

AUTOID is the default if USRID is not specified.

USRID is optional and is written on the same line as TARGET and BLK. It supersedes AUTOID and associates the specified user with the group of requests defined for that BLK parameter; for example:

```
TARGET=DB2A , BLK=BLKDB2A , USRID=$TDB2AR
```

If USRID is not specified, the default is AUTOID. If AUTOID and USRID are not specified, a default user ID of USERID is used. A USERID member must exist in the BBPARM data set.

- Step 2** Identify the name of the data set used for a trace directory (required by MAINVIEW for DB2, MAINVIEW for IMS Online, MAINVIEW for DBCTL, and MAINVIEW for CICS for trace logging).

Specify the data set name of the trace directory as:

```
TRDIR=dsn, SUBSYS=ssid
```

where:

dsn Is a valid data set name.

ssid Is the subsystem name to which this trace directory applies (optional). This parameter allows multiple BBI-SS PASs to share the same BBIISP00 member.

Trace logging can be set up automatically using the AutoCustomization procedures described in the *OS/390 and z/OS Installer Guide*. Your product's *Customization Guide* describes how to set up and maintain trace logging manually.

Step 29: (Optional) Set BBI-SS PAS Historical Recording Interval

This section applies only to Group 3 products. See Table 4-1 on page 4-2 for a list of products in this group.

BBPARM Member BBIISP00:

Use your copy of this member to specify a recording interval when records are to be written to a historical data set.

Records are written from the interval recorder to a historical data set at the end of every interval recorder recording interval (IRRI). The `IRRI=` statement specifies the length of the recording interval.

Views displaying current data have interval and realtime data collection periods based on the IRRI value.

Note: If the `IRRI=` statement is not specified in member BBIISP00, the length of the recording interval defaults to 15 minutes.

Edit member BBIISP00 to:

Step 1 Add the `IRRI=` statement to BBIISP00.

Step 2 Specify a recording interval in hours and minutes (*hh:mm*) for the `IRRI=` statement.

For example, to set the historical interval recording interval to 10 minutes, specify:

```
IRRI=00:10
```

The shortest recording interval you can specify is two minutes, `IRRI=00:02`. The longest recording interval you can specify is one hour, `IRRI=01:00`.

Step 30: (Optional) Allocate BBI-SS PAS Journal and Image Logs

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to allocate log and profile data sets used during execution of a BBI-SS PAS.

BBSAMP Member SSLOG:

This member contains sample JCL to allocate BBI-SS PAS journal and image log data sets. For MAINVIEW AutoOPERATOR, it contains a sample statement to allocate a profile variable pool called BBIVARS for MAINVIEW AutoOPERATOR EXECs.

Notes:

- Do not alter the log attributes because the defaults are required and are reset automatically if you change them.
- If you are using products on multiple systems and want to use the same JCL procedure for all of them, use the SSID in the data set name of the logs. All the subsystems can then use the same PROC.
- BBI-SS image log allocation is not used by MAINVIEW AutoOPERATOR or MAINVIEW for WebSphere MQ. It is recommended for MAINVIEW for CICS, MAINVIEW for IMS Online, MAINVIEW for DB2, and MAINVIEW for DBCTL.

To allocate the BBI-SS PAS journal and image log data sets:

Step 1 Globally change the following (including the S1, S2, and S3 statements at the end of this job):

- Change the PREFIX parameter to the prefix used to load the product libraries to DASD.
- Set the SS parameter value to the subsystem ID of the BBI-SS PAS.
- Set the DV parameter value to the volume serial of the allocation volume for the BBI-SS PAS journal data sets.

- Set the DU parameter value to the allocation volume unit type.

Step 2 If you do not have MAINVIEW for CICS, MAINVIEW for IMS Online, MAINVIEW for DBCTL, or MAINVIEW for DB2, comment out the ALLOCIMG (S1) procedure.

Step 3 If you do not have MAINVIEW AutoOPERATOR, comment out the ALLOCBVP (S3) procedure.

If you have MAINVIEW AutoOPERATOR, the ALLOCBVP statement is required to allocate the BBIVARS profile variable pool.

Step 31: (Required) Create a BBI-SS PAS Start Procedure

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to set up a started task procedure for the BBI-SS PAS.

BBSAMP Member SSJCL:

Use this member to help you set up your BBI-SS PAS started task procedure.

Notes:

- The default region size for the BBI-SS PAS is 64M.

If you reduce this allocation and there is not enough working storage available after initialization, the BBI-SS PAS terminates with an appropriate message.

- SSJCL should run as a started task; if not, a JES initiator is occupied for the duration of the BBI-SS PAS session.

To run SSJCL as a started task, remove the JOBCARD statement and the PEND and EXEC statements at the end of the JCL. Copy the SSJCL JCL procedure to a JES PROCLIB.

- For optimum BBI-SS PAS performance, the BBI-SS PAS MVS dispatching priority should be equal to or greater than any of the target subsystems it manages.

Running with a dispatching priority that is too low could result in loss of data or excessive storage usage.

- If you have RACF or an equivalent authorization, verify that the BBI-SS PAS has authorization to access and update all MAINVIEW product data sets.
- The PARM string defines the subsystem ID assigned to the BBI-SS PAS.

Step 1 Change the PREFIX parameter to the prefix used to load the product libraries to DASD.

Step 2 Set the SS parameter value to the subsystem ID of the BBI-SS PAS.

Step 3 Use the default START=WARM.

START=COLD should be used only as instructed by BMC Software Customer Support or when the BBI-SS PAS requests it through WTOR after maintenance.

Step 4 Review your parameter library allocation to ensure that the UBBPARM data set is concatenated before the BBPARM (see page 3-6) data set.

Step 5 For MAINVIEW AutoOPERATOR and MAINVIEW for WebSphere MQ, if you plan to use any of the following: :

- MAINVIEW AutoOPERATOR COMMAND/POST Extension (ACE)
- MAINVIEW AutoOPERATOR Command MQ On Ramp (COR)
- MAINVIEW AutoOPERATOR Automation Power Line (APL)

or

- If you connect a BBI-SS PAS from MAINVIEW for WebSphere MQ *without* using MQSeries channels to Command MQ for D/S

complete this step to enable the BBI-SS PAS to use IBM's MVS/ESA TCP/IP 3.1 product or the BMC Software Generic TCP/IP Server (GTS), as applicable.

5.A Add the following TCP/IP libraries to the STEPLIB:

```
tcprefix.SEZATCP  
tcprefix.SEZALINK
```

where `tcprefix` was defined when you installed TCP/IP.

MAINVIEW for WebSphere MQ requires the following additional library to be added to STEPLIB:

```
tcprefix.SEZALPA
```

where `tcprefix` was defined when you installed TCP/IP.

5.B Add the following TCP/IP configuration DD statement to the SSJCL:

```
//SYSTCPD DD DISP=SHR,DSN=tcpprefix.CNTL(tcipdata)
```

where `tcpprefix` was defined when you installed TCP/IP and `tcipdata` is the member name that contains the TCP/IP configuration information required by TCP/IP client programs.

5.C Required for MAINVIEW AutoOPERATOR and MAINVIEW for WebSphere MQ only)

Add BMC Software Generic TCP/IP Server (GTS) support in the BBI-SS PAS

In BBPARM member BBISSPxx, specify:

```
GTS=xx
```

where `xx` is the suffix of your BBTTCPxx member.

5.D (Required for MAINVIEW for WebSphere MQ; optional for MAINVIEW AutoOPERATOR.)

Create BBPARM member BBTTCP00 (refer to “BBPARM Member BBTTCP00 Parameters” on page C-9 for more information.)

If you chose not to complete this step, specify `GTS=00` in BBPARM member BBISSPxx (see Step 5.C).

Create BBPARM member BBTTCPxx in your copy of the BBPARM library where `xx` is 00 or, the suffix you specify with the `GTS=` parameter in BBPARM member BBISSPxx.

BBPARM member BBTTCP00 defines your TCP/IP environment and has the following statements:

```
TCPNAME=TCP/IP started task
STACK=IBM32 | ILINK52
```

where:

`TCPNAME` is set to the started task name of your TCP/IP job.

`STACK` is set based on the version of TCP/IP you are using. If you are running TCP/IP V.3.2 or higher, use `IBM32`. If you are using Interlink TCPAccess 5.2, use `ILINK52`.

BMC Software recommends that you not to specify any parameters in member BBTTCpxx. The parameters are designed mostly for testing or support purposes. When parameters are not specified, the BBI-SS PAS chooses the most current TCP/IP stack which is found active on the system.

Note: If you are starting the BBI-SS PAS address space before TCP/IP, GME will not activate during BBI-SS initialization. Both GTS and GME must be activated after TCP/IP is started. You can do this by issuing the following two BBI commands:

```
.S GTS  
.E P AAOGMExx,RECYCLE
```

Step 6 MAINVIEW AutoOPERATOR only

- **AutoOPERATOR EXECs:** A SYSPROC DD statement is required. It points to the libraries that contain MAINVIEW AutoOPERATOR EXEC members.

— Uncomment the SYSPROC DD statement.

— Review your data sets allocated by SYSPROC to ensure that the UBBPROC data set is concatenated before the BBPROC data set.

- **AutoOPERATOR Data Storage:** BBIDIV is a mandatory data set that is used to store binary large objects and data across MAINVIEW AutoOPERATOR restarts.

— Insert the following DD statement immediately before the PEND statement:

```
//BBIDIV DD DSN=&PREFIX..&SS..BBIDIV,DISP=OLD SS
```

Do not specify the BBIDIV DD statement using a DUMMY (null) data set. This might cause an S013-64 abend.

— See “Step 32: (Required) Allocate MAINVIEW AutoOPERATOR Data Sets” on page 5-64 for information about how to allocate the BBIDIV data set.

Step 7 (Optional) Add the following statement to your configuration members:

```
//BBCFG DD DSN=&PREFIX..&SS..UBBPARM (xxxxxxxx),DISP=SHR  
SS
```

where *xxxxxxxx* is a 1- to 8-character name of the member you created for BBCFG in “Step 27: (Optional) Define Configuration Members to BBI-SS PAS” on page 5-51

Step 32: (Required) Allocate MAINVIEW AutoOPERATOR Data Sets

This section applies only to MAINVIEW AutoOPERATOR.

This step describes how to allocate the BBIDIV data set required for MAINVIEW AutoOPERATOR.

Note: BBIVARS, described in “Step 30: (Optional) Allocate BBI-SS PAS Journal and Image Logs” on page 5-57, is also a required MAINVIEW AutoOPERATOR data set.

BBSAMP Member DIVDEF:

Use this member to allocate a BBIDIV data set (see “Step 31: (Required) Create a BBI-SS PAS Start Procedure” on page 5-59), which is required to operate MAINVIEW AutoOPERATOR. This data set is used to store binary large objects and data across restarts. The BBI-SS PAS must be down when this data set is allocated. Each BBI-SS PAS must have its own copy of this data set; it cannot be shared.

Step 1 Modify the job statement to your installation’s requirements.

Make the following changes to the procedure header:

- Change *HILEVEL* to the high-level qualifier for MAINVIEW product data sets.
- Change *SSID* to the 4-character subsystem name.
- Change *VOL* to the volume serial number of the DASD volume where the linear data set will be allocated.

Step 2 Submit the newly modified job.

Step 33: (Required) Create Automatic BBI-SS PAS Log Handling Procedures

This section applies to MAINVIEW Alarm Manager and the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

This step describes how to create an automatic BBI-SS PAS journal log print and reset procedure and an image log print procedure.

Note: MAINVIEW Alarm Manager does not use an image log; it uses only a journal log. If you are performing this step for MAINVIEW Alarm Manager, you can skip the information related to ILOGJCL and the image log.

BBPARM Member BBISSPxx:

Use this member to specify started task procedures for automatic journal and image log handling.

BBSAMP Member DLOGJCL:

Use DLOGJCL to set up a started task to print and reset the BBI-SS PAS journal log.

BBSAMP Member ILOGJCL:

Use ILOGJCL to set up a started task to print the BBI-SS PAS image log if your products use this image log.

Step 1 Automatic BBI-SS journal log print and reset procedure

When the journal log is full, a SWITCH JOURNAL command is issued and a user exit named BBIUSR01 is invoked. If an I/O error is encountered on the journal, it causes a journal switch and the same user exit, BBIUSR01, is invoked. This exit invokes a started task that prints and resets the journal log. If there is a journal I/O error, an I/O error message is issued before the started task is invoked.

The started task name is specified with the following JRNLSTC statement in BBISSPxx:

```
JRNLSTC=xxxxxxxx
```

where xxxxxxxx is a 1- to 8-character name of a started task used to print and reset the BBI-SS PAS journal logs. BBIDLOG is the default. The value you enter for xxxxxxxx must match the name of the started task that you add to your JES PROCLIB (described in the steps that follow).

Copy BBSAMP member DLOGJCL into your JES PROCLIB data set and rename the member to the value you specified for JRNLSTC in BBISSPxx. Make the following changes in the JCL:

- Change the name BBIDLOG in the PROC statement to the name you selected for your print/reset procedure name.
- Change all occurrences of BBIID to SSID.
- Change the PREFIX= 'xxxxxx' parameter in the PROC statement to the high-level qualifier used to load the product libraries to DASD.

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). Any suitable BLKSIZE is allowed.

The journal log print/reset procedure is ready for use. If the product is already running when this step is completed, the BBISSPxx parameter change does not take effect until the next BBI-SS PAS start.

Note: The source for the BBIUSR01 user exit is distributed in the BBSAMP data set and can be modified. BBSAMP member ASMLKED contains sample JCL you can use to assemble and link-edit the BBIUSR01 user exit after modification. However, modifying BBIUSR01 is not necessary since it is also distributed in module form. Any changes made to this exit are the responsibility of the user. BMC Software Customer Support can offer assistance only if the exit does not operate as described.

Step 2 Automatic BBI-SS PAS image log print procedure

When the image log is full, a SWITCH IMAGE command is issued and a user exit named BBIUSR02 is invoked. If an I/O error is encountered on the image log, it causes a log switch and the same user exit, BBIUSR02, is invoked. This exit invokes a started task that prints the image log. If there is an image log I/O error, an I/O error message is issued before the started task is invoked.

The started task name is specified with the following IMAGSTC statement in BBISSP xx :

```
IMAGSTC=xxxxxxxx
```

where $xxxxxxxx$ is a 1- to 8-character name of a started task used to print the BBI-SS PAS image logs. BBILOG is the default. The value you enter for $xxxxxxxx$ must match the name of the started task that you add to your system PROCLIB (described in the steps that follow).

Copy BBSAMP member ILOGJCL into your JES PROCLIB data set and rename the member to the value you specified for IMAGSTC in BBISSP xx . Make the following changes to the JCL:

- Change the name BBILOG in the PROC statement to the name you selected for your print procedure name.
- Change all occurrences of BBIID to SSID.
- Change the PREFIX= ' $xxxxxx$ ' parameter in the PROC statement to the high-level qualifier used to load the product libraries to DASD.

The image log print procedure is ready for use. If the product is already running when this step is completed, the BBISSP xx parameter change does not take effect until the next BBI-SS PAS start.

Note: The source for the BBIUSR02 user exit is distributed in the BBSAMP data set and can be modified. BBSAMP member ASMLKED contains sample JCL you can use to assemble and link-edit the BBIUSR02 user exit after modification. However, modifying BBIUSR02 is not necessary since it is also distributed in module form. BMC Software does not support any user modifications made to BBIUSR02.

Step 34: (Optional) Define Terminal Session (TS) Parameters

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3 (except for MAINVIEW for WebSphere MQ)

See Table 4-1 on page 4-2 for the list of products in each group. These products use the parameters in BBPROF members BBITSP00 and xxxxxPFK when the TS initializes.

Instead of manually customizing this member, you can use Option 0 from the MAINVIEW Selection Menu to set up BBITSP00. Option 0 also allows you to select the TS options for all your other MAINVIEW products.

This step describes how to define TS target system and primary product application defaults and create unique TS user profiles.

BBPROF Member BBITSP00:

Use this member to set up your TS user profiles, target system, and primary product application.

Step 1 Create a user profile:

userid.BBPROF

where *userid* is the user's TSO logon ID. The user profile can be concatenated in front of the site's common BBPROF data set.

Users should have their own BBPROF data sets so that each can specify:

- Unique PF keys
- CYCLE commands
- Target system defaults
- Primary Option Menu
- A unique set of application profiles
- User graph members (MAINVIEW for CICS)

The MAINVIEW CLIST creates a user BBPROF automatically with a PROFILE(YES) statement. If PROFILE(NO) is specified, the BBPROF data set is shared.

Note: You can also create a unique site-specific profile as SBBPROF. If you use a common, site BBPROF data set, profile members for product applications are automatically updated by the last person who used that application.

Step 2 In BBITSP00, specify the following TS parameters:

```
TARGET=tgtname ,TYPE=appltype [ ,PRIMMENU=pnlname ]
PRIMAPPL=appltype
ISPF=YES | NO
[ BBIRESP=LOCAL ]
[ IMSRESP=LOCAL ]
```

where:

TARGET

Specifies a 1- to 8-character target name.

For MAINVIEW for CICS, specify a CICS target.

For MAINVIEW for DB2, specify a DB2 target (subsystem ID).

For MAINVIEW for IMS Online or MAINVIEW for DBCTL, specify an IMS or DBCTL target.

For MAINVIEW AutoOPERATOR, specify an IMS, CICS, or MVS target.

TYPE

Specifies the product related to the target specified in the TARGET parameter, as one of the following:

AO	For MAINVIEW AutoOPERATOR
CICS	For MAINVIEW for CICS
IMS	For MAINVIEW for IMS Online and MAINVIEW for DBCTL
DB2	For MAINVIEW for DB2

PRIMAPPL	Specifies the primary application to be displayed when the TS starts. Specify one of the following:
AO	MAINVIEW AutoOPERATOR
CICS	MAINVIEW for CICS
IMS	MAINVIEW for IMS Online and MAINVIEW for DBCTL
DB2	MAINVIEW for DB2
NONE	The MAINVIEW Primary Option Menu is displayed after logging on to the TS.

To display the same Primary Option Menu for multiple targets, list the TARGET statements followed by the PRIMAPPL statement, for example:

```
TARGET=DETROIT , TYPE=AO
TARGET=IMS22X , TYPE=IMS
TARGET=DB2P , TYPE=DB2
PRIMAPPL=DB2
```

If PRIMAPPL is not specified, the Primary Option Menu lists all available products when the TS starts.

PRIMMENU	By default, a Primary Option Menu is selected for the product or combination of products you have installed for a particular target. If you have performance management products and the MAINVIEW AutoOPERATOR option for IMS or CICS installed, the default menu shows that option and the performance management products. If you want to limit access to MAINVIEW AutoOPERATOR, PRIMMENU can be specified as follows:
----------	--

```
TARGET=IMS22K , TYPE=IMS , PRIMMENU=IMS@MVMR
TARGET=CICSA1 , TYPE=CICS , PRIMMENU=CICS@MR
```

The default for PRIMMENU is determined by the products installed. The values for PRIMMENU can be:

IMS@MVA	(MAINVIEW for IMS Online, MAINVIEW for DBCTL, and MAINVIEW AutoOPERATOR for IMS) The Primary Option Menu for the MAINVIEW for IMS Online performance management products and MAINVIEW AutoOPERATOR for IMS (if installed).
IMS@MVMR	(MAINVIEW for IMS Online and MAINVIEW for DBCTL) The Primary Option Menu for the performance management components (Resource Analyzer, Resource Monitor, Workload Analyzer, and Workload Monitor)
DB2@MVMR	(MAINVIEW for DB2) The Primary Option Menu for all the MAINVIEW for DB2 product services.
DB2@RXnn	(MAINVIEW for DB2) The Primary Option Menu for all the MAINVIEW for DB2 product services, including an option to access RxD2, where <i>nn</i> represents the MAINVIEW for DB2 release number.
CICS@MVA	(MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS) The Primary Option Menu for the MAINVIEW for CICS performance management services and MAINVIEW AutoOPERATOR for CICS (if installed).
CICS@MR	(MAINVIEW for CICS) The Primary Option Menu for MAINVIEW for CICS performance management services.
ISPF	Specify YES or NO. YES initializes the product in an ISPF window. NO initializes the product under TSO. The default is ISPF=YES.
BBIRESP	(MAINVIEW AutoOPERATOR) LOCAL returns responses to BBI control commands issued from a TS to your TS journal and BBI-SS PAS journal. If BBIRESP is not used, BBI control command responses are written to the BBI-SS PAS journal only.
IMSRESP	(MAINVIEW AutoOPERATOR for IMS) LOCAL returns IMS command responses to your TS journal.

Step 3 BBPROF Member BBICMDS: BBI Command Table

If your BBPROF data set does not contain the BBICMDS member compatible with the current TS version, you may see an error message advising you that your BBICMDS member should be at a more recent PTF level. Check your data set concatenation. BBICMDS should not be in any data set concatenated in front of the distributed BBPROF data set.

Step 35: (Optional) Define Cross-System Communication

This section applies to all MAINVIEW products.

This step describes how the MAINVIEW common communications framework described in Chapter 1, “Customizing MAINVIEW” and shown in Figure 1-1 on page 1-2 is set up. This allows a single TS, using one or more MAINVIEW products, to monitor and manage multiple local or remote targets (MVS itself in a sysplex or non-sysplex environment or subsystems like CICS, IMS, and DB2).

A TS connects to:

- A CAS if there is one available

A CAS communicates with other CASs on other local and remote systems and allows direct communication between an individual TS and a product address space. Usually, there is one CAS per MVS system image, but there is no limit to the number of remote systems with CASs with which a single CAS can communicate.

- A BBI-SS PAS

BBI-SS PASs on local and remote systems are linked together to provide cross-system communication for an individual TS through a local BBI-SS PAS to any other BBI-SS PAS.

Depending on the products installed, the BBI-SS PAS may also connect to a CAS on that MVS image.

The instructions in this step explain how to set up:

- CAS to CAS communication
- BBI-SS PAS to BBI-SS PAS communication

shown in Figure 5-3 on page 5-73.

When CASs reside in the same sysplex, CAS to CAS communication through XCF (Coupling Facility), as shown in Figure 5-4 on page 5-74, is automatically established.

Figure 5-3 Define MAINVIEW Cross-System Communication

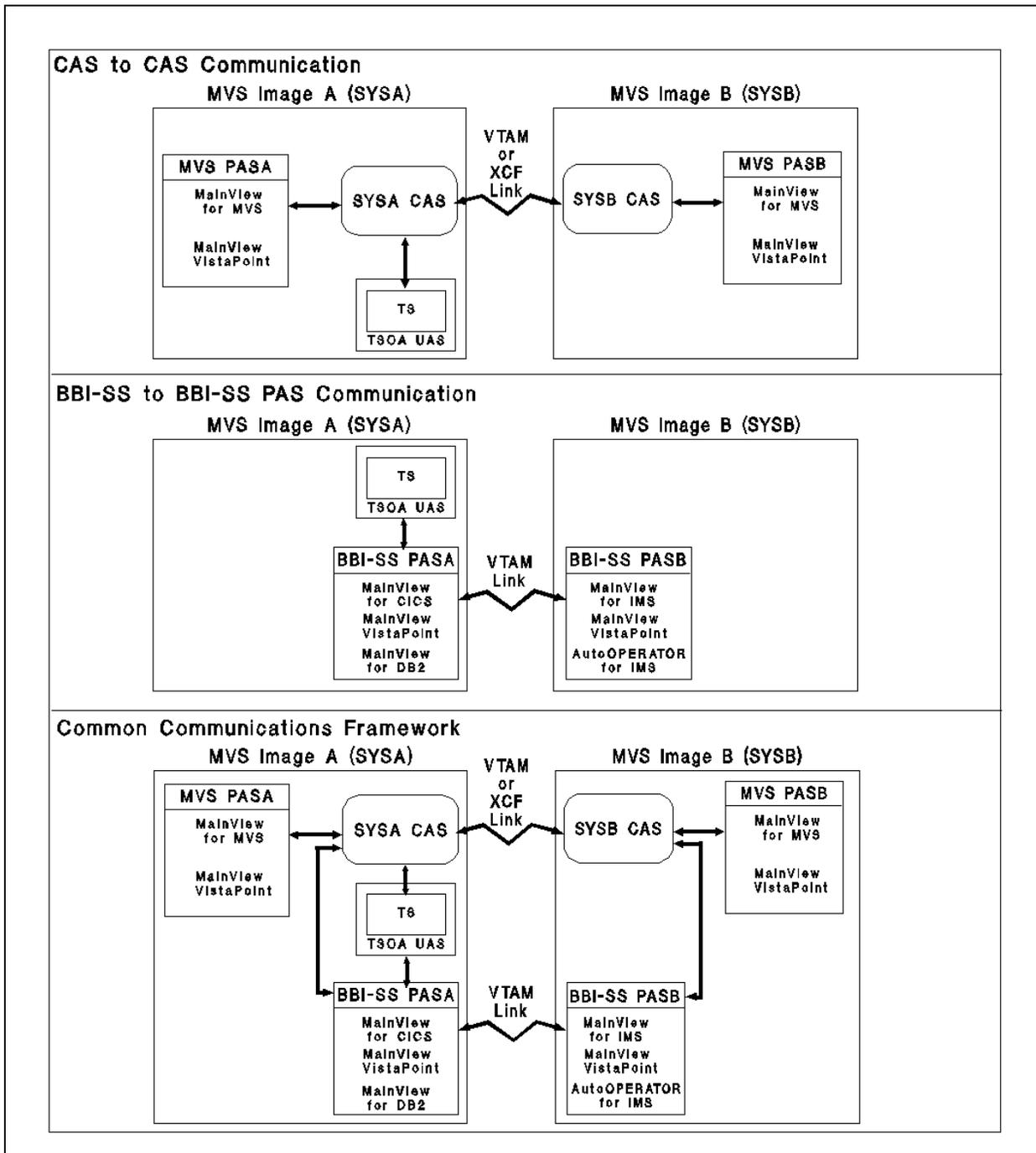
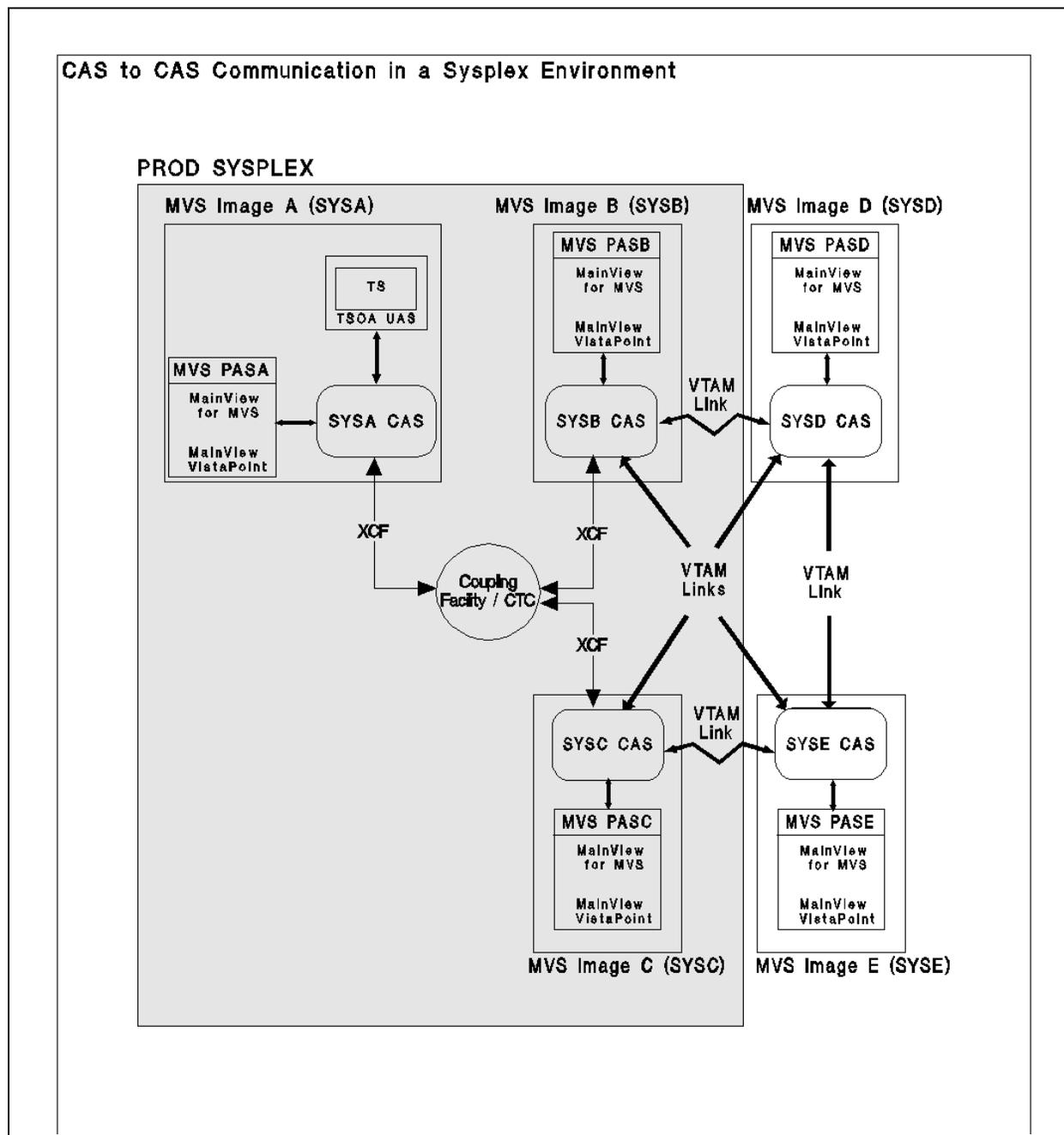


Figure 5-4 MAINVIEW Cross-System Communication in a Sysplex Environment



Plan for CAS to CAS Communication

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

Note: If you created VTAM definitions for a previous release, you do not need to do this step unless you are running a previous release and a new release on the same system.

Follow the instructions in “Define CAS Definitions for the First Time” on page 5-95 to migrate your existing CAS definitions.

You need to create separate VTAM definitions for:

- Products running in the MVS PAS as described in “Define a CAS to CAS Communication” on page 5-86
- Products running in the BBI-SS PAS as described in “Define BBI-SS to BBI-SS PAS Communication” on page 5-82.

Before CASs can request information from each other through XCF or VTAM, local and remote CAS definitions must be defined to each CAS. You may need to change or delete these definitions to maintain your cross-system communication properly, as described in this section.

A CAS definition is uniquely named and contains:

- Identification information about a CAS
- An XCF group name (default is BBGROUP) used by a CAS to establish a communication link with other CASs residing in the same sysplex

The XCF member name is the same as the CAS name.

- A VTAM major node name

This is optional. It is needed only if a CAS is going to communicate with other CASs residing in a different sysplex.

- Optional parameter member suffixes to control cross-system communication in a shared DASD environment

CAS Cross-System Communication Considerations

Before you begin defining your CASs and cross-system communication links as described in this section and in “Define BBI-SS to BBI-SS PAS Communication” on page 5-82, you should:

Step 1 Determine whether or not the CAS is running in a sysplex environment.

In a sysplex environment (see Figure 5-4 on page 5-74), a CAS automatically communicates with other CASs running in the same sysplex through XCF (Cross-System Coupling Facility). A VTAM definition is not required for this environment. If one exists, it is used for backup.

Step 2 Determine which systems you want the current CAS to communicate with (see “CAS Communication Links” on page 5-77 and “CAS Cross-System Customization in a Shared Environment” on page 5-78).

If the other CASs are not in the sysplex (see Figure 5-4 on page 5-74), then a VTAM definition is needed for the current CAS as described in “Define a CAS to CAS Communication” on page 5-86.

Step 3 Obtain the MVS system names from SYS1.PARMLIB member IEASYSxx for each system that is to establish cross-system communication.

For MVS/ESA, Version 5, for example, the parameter is SYSNAME.

Step 4 Ensure that your user ID is authorized to access each MVS system.

Step 5 Obtain the VTAM CDRM name for each MVS system outside the sysplex from SYS1.VTAMLST(CDRMxx).

Step 6 Determine a VTAM major node name for each CAS on each MVS system.

This is required only if the current CAS must communicate with other CASs outside the same sysplex.

Note: In a shared DASD environment, you can define one major node name for all your CASs if the minor node names are unique throughout the sysplex or multisystem environment.

Step 7 Determine a VTAM minor node name for each CAS not in a sysplex.

Step 8 Ensure that your user ID is authorized to modify the VTAM configuration for each MVS system. If you are not authorized, you may need assistance from your VTAM administrator.

Step 9 You should use the MVS system name as the name for each CAS, since only one CAS should run per MVS system; if you cannot use the MVS system name, then determine a CAS name for each system.

Note: If you are running products at different release levels on the same system, such as using a production system to run test or trial products at a newer level than products from a previous level, then a unique CAS name is required.

- Step 10** If any of your MVS systems are running in sysplex mode, obtain the sysplex name for each MVS system from SYS1.PARMLIB member (COUPLxx).
- Step 11** For CMF MONITOR customers requiring XCF communication links, determine if you can use the default BBGROUP as the XCF group name. If you cannot, determine a valid group name for XCF communication. (To understand how to customize the XCF communication group name, see “Customize An XCF Group Name” on page 5-94.)

CAS Communication Links

CAS to CAS communication allows:

- Data to be shipped back and forth between systems
- Users to open windows and obtain data from both systems

CAS to CAS communication requires:

- Security customization to prohibit access for some users, if necessary
- Creation of VTAM definitions for a CAS only if that CAS is going to communicate with other CASs in a different sysplex
- Customization of CAS definitions for each system

The communication link between two CASs can be through VTAM or XCF (see Figure 5-4 on page 5-74):

- XCF link:

CASs running on different MVS images within the same sysplex can communicate directly through an XCF link. An XCF communication link is automatically established between two CASs when they have the same XCF group name in the CAS definition (see “Customize An XCF Group Name” on page 5-94). A VTAM link is optional and used only for backup.

- VTAM link:

CASs that are not in the same sysplex or in a different CAS XCF group can communicate via a VTAM link as described in “Define a CAS to CAS Communication” on page 5-86. A VTAM link is not usually used between CASs in the same sysplex. If one exists, it is used for backup when an XCF link is not available.

If you have some CASs that you do not want to allow access to all others, you can:

- Define a security definition using the SERDEF view (described in *Implementing Security for MAINVIEW Products*) that prevents user access to a cross-system CAS and its data
- Define a separate parameter library for the CAS you want to restrict

CAS Cross-System Customization in a Shared Environment

If the systems are running in a shared DASD environment and the CASs are accessing a shared parameter library:

- VTAM application major nodes and CDRM definitions that all reside in the same SYS1.VTAMLST library can be defined in a single member, so only one system configuration list need be updated.

However, if only one major node member is created, the minor node name defined for each CAS must be unique throughout your sysplex or multisystem environment.

If SYS1.VTAMLST is not shared, a separate VTAM major node must be created (see “Define a VTAM Major Node for the CAS” on page 5-87 for more information) and each system's configuration list must be updated (see “Update the VTAM Configuration List” on page 5-92 for more information).

- CAS definitions do not have to be defined repeatedly in each CAS.

You only need one CAS definition parameter library member containing all the definitions for all the CASs that are to communicate with one another. Each CAS points to this library with its CAS procedure BBIPARM DD statement (see “BBILIB member @@YZZ021 contains statements for creating a CAS startup procedure in your system started task procedure library. These statements are described in the following table.” on page A-1). This avoids having to create definitions for a local and cross-system CAS on each system individually. It allows you to define one set of definitions in one CAS's CASDEF view (see “Define CAS Definitions for the First Time” on page 5-95) and use only the **INSTall** command in all the other CASs' CASDEF views.

If you have certain CASs that you do not want to communicate, such as the test system CASs and production system CASs, then you must have one BBIPARM library for the test system CASs and a different BBIPARM library for the production system CASs.

“Define CAS Definitions for the First Time” on page 5-95 explains how to define CAS definitions in detail.

CAS Cross-System Communication Requirements

For communication between CASs in the same sysplex:

- The default XCF group name of BBGROUP automatically provides a communication link between CASs in the same sysplex.

All CASs must have the same XCF group name in the CAS definition. As long as the default group name is not changed, CASs in different MVS images can communicate directly through the XCF link.

- No VTAM definition is needed.

If a VTAM definition exists, it is used for backup when the XCF link is not available.

Communication between CASs in different sysplexes requires:

- A VTAM application major node definition to identify itself to VTAM

A VTAM application major node definition identifies a CAS to VTAM. Once VTAM knows about a CAS, the CAS can issue instructions to VTAM to establish communication links with other CASs that other VTAMs know about.

- VTAM CDRM (Cross-Domain Resource Manager) definitions to identify the cross-system CASs known to other VTAMs

A VTAM CDRM definition identifies other CASs a single CAS can communicate with. VTAM must know about all the cross-system CASs to establish connections between them.

- CAS definitions on each system for CAS communication through VTAM

as described in “Define a CAS to CAS Communication” on page 5-86.

Plan for BBI-SS PAS to BBI-SS PAS Communication

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group. If you only have a product in Group 2, go to “Define a CAS to CAS Communication” on page 5-86.

Before you begin defining your PASs as described in “Define BBI-SS to BBI-SS PAS Communication” on page 5-82, you should:

- Step 1** Determine which systems you want the current PAS to communicate with.
- Step 2** Determine a VTAM major node name and minor node name for each PAS on each MVS system.
- Step 3** Ensure that your user ID is authorized to modify the VTAM configuration for each MVS system. If you are not authorized, you may need assistance from your VTAM administrator.

Add a BBI-SS PAS

You can add a new BBI-SS PAS so that:

- More than the allowable number of target systems (see Step 3 on page 5-47) can be managed.
- A different target system can be managed; for example, there could be one BBI-SS PAS connected to a test system and another connected to a production system.
- Unique specifications for a BBI-SS PAS can be defined; for example, for security requirements.
- TSs can be processed by one MVS and the target regions can be processed by another MVS.
- A BBI-SS PAS can communicate with another BBI-SS PAS.

To add a new BBI-SS PAS, you must set up each one first and then define their interconnections if you want BBI-SS to BBI-SS PAS communication:

- Step 1** Define the BBI-SS PAS and its associated target in BBPARM member BBIJNT00, as described in “Step 26: (Required) Define BBI-SS PAS Suffixes and Target System Parameters” on page 5-46, and BBPARM member BBISSPxx.
- Step 2** Define the connections to any existing BBI-SS(s) that you want to communicate with the new BBI-SS (see “Define BBI-SS to BBI-SS PAS Communication” on page 5-82).
- Step 3** If a new BBI-SS PAS is connected to an active BBI-SS PAS, the active BBI-SS PAS must be restarted to activate the connection.

BBI-SS PAS Communication Links BBI-SS PAS

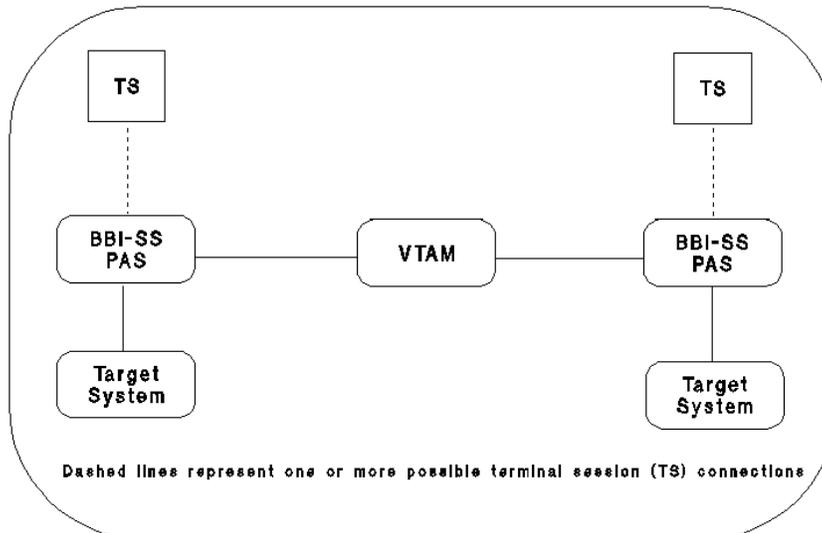
Subsystem communication allows a user from a single TS to manage one or more target systems that belong to multiple BBI-SS PASs. The BBI-SS PAS processes communication between one or more TSs and one or more targets when:

- A TS is connected to one BBI-SS PAS.
- Multiple TSs are connected to a single BBI-SS PAS.
- Multiple target systems are assigned to a single BBI-SS PAS.
- A TS can access data from multiple targets through its associated BBI-SS PASs.
- A request from a TS can be routed by its associated BBI-SS PAS to another BBI-SS PAS automatically.

“Define BBI-SS to BBI-SS PAS Communication” on page 5-82 describes the VTAM parameters you must define for single domain and cross-domain communication. VTAM parameters are required by the VTAM system programmer to correctly define the MAINVIEW communication environment. For additional information about the VTAM parameters, see the IBM publication *Advanced Communications Function for VTAM - Installation and Resource Definition*.

All BBI-SS PAS to BBI-SS PAS communication is routed through VTAM, as shown in Figure 5-5 on page 5-82.

Figure 5-5 BBI-SS PAS to BBI-SS PAS Communication



Define BBI-SS to BBI-SS PAS Communication

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

This section describes how to manually define subsystem communication for BBI-SS PASs in a single domain and expand the definition to provide for communication with a BBI-SS PAS in another domain. The description uses an example of setting up BBI-SS PAS communication with two subsystems, named BBIA and BBIB, which could be in either a single domain or cross domain. The example first shows how to connect two BBI-SS PASs in the same domain. It then explains how to define BBI-SS PAS cross-domain communication.

Define VTAM Major Node for BBI-SS PAS

The instructions in this section use an example of two BBI-SS PASs, named BBIA and BBIB, to show you how to create BBI-SS PAS definitions for cross-system communication. A BBI-SS PAS, named BBIA, monitors one target system and another BBI-SS PAS, named BBIB, monitors another target system. To establish communication between these two subsystems:

- Both must be defined as BBI subsystems
- A VTAM communication path between them must be defined

Note: Review the MAXDATA parameter in BBPARM member BBIXSP00 for compatibility with the VTAM MAXDATA parameter.

The MAXDATA parameter of the NCP PCCU macro for the Multisystem Networking Facility (MSNF) links specifies the maximum size buffer for VTAM. The MAXDATA parameter in BBIXSP00 specifies the maximum size buffer that BBI passes to VTAM. VTAM adds a transmission header to the BBI buffer before including it in its buffer. The BBI MAXDATA value, therefore, must be 50 bytes less than the VTAM MAXDATA value.

Subsystem names and their associated targets are defined in shared BBPARM members BBINOD00 and BBIJNT00. BBINOD00 defines the BBI-SS PASs for subsystem communication and BBIJNT00 defines targets and their associated BBI-SS PAS. For example, the parameters to define subsystems BBIA and BBIB in the node table are:

```
BBINOD00  SUBSYS=BBIA , APPLID=$BBIA
            SUBSYS=BBIB , APPLID=$BBIB
```

where:

BBIA and BBIB are the subsystem IDs specified in the BBIA and startup JCL.

\$BBIA and \$BBIB are the same names used in the VTAM application node definitions (see “Create Cross-Domain BBI-SS PAS Resource Manager Definitions” on page 5-83).

```
BBIJNT00  TARGET=CICSA , TYPE=CICS , SUBSYS=BBIA
            TARGET=CICSB , TYPE=CICS , SUBSYS=BBIB
```

Create Cross-Domain BBI-SS PAS Resource Manager Definitions

The subsystem communication path is defined through VTAM. The path can be single domain or cross domain. The single-domain path is defined as a VTAM application node. The cross-domain path is additionally defined as a VTAM cross-domain resource.

Single Domain:

If BBI-SS PASs BBIA and BBIB are in the same domain, as shown in Figure 5-6 on page 5-84, the following VTAM application nodes are defined:

SYS1.VTAMLST(BBIAPPLA)

```

                VBUILD  TYPE=APPL
$BBIA          APPL    AUTH=( ACQ ,NOTSO ,VPACE )
    
```

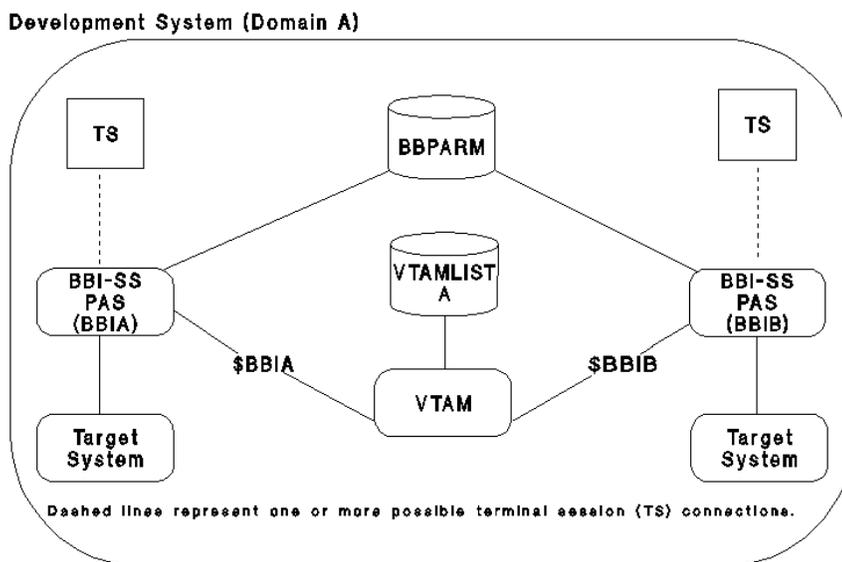
SYS1.VTAMLST(BBIAPPLB)

```

                VBUILD  TYPE=APPL
$BBIB          APPL    AUTH=( ACQ ,NOTSO ,VPACE )
    
```

Note: The two \$BBIA and \$BBIB nodes can be defined within one major node, such as BBIAPPLA. You can choose names that conform to your site standards, but they must be the same as the APPLID parameters specified in BBPARM member BBINOD00 for the respective BBI-SS PASs. This example uses the sample member names (BBIAPPLA, BBIAPPLB, BBIXSA, BBIXSB) distributed by BMC Software in the BBSAMP data set.

Figure 5-6 BBI-SS PAS to BBI-SS PAS Single-Domain Communication



If BBI-SS PASs BBIA and BBIB are cross domain (see Figure 5-7 on page 5-86), then VTAM cross-domain resources must be defined; for example:

Cross Domain A Definitions:

Define an application node (BBIAPPLA) and a cross-domain resource (BBIXSA) in Domain A as follows:

SYS1.VTAMLST(BBIAPPLA)

```

                VBUILD  TYPE=APPL
$BBIA          APPL    AUTH= ( ACQ , NOTSO , VPACE )

```

SYS1.VTAMLST(BBIXSA)

```

                VBUILD  TYPE=CDRSC
$BBIB          CDRSC   CDRM=cdrmname , ISTATUS=ACTIVE

```

where:

\$BBIB Is the cross-domain node for the BBI-SS PAS used in the VTAM application node definition in Domain B.

cdrmname Is the name of the CDRM major node defined for Domain B.

Domain B Definitions:

Define an application node (BBIAPPLB) and a cross-domain resource (BBIXSB) in Domain B as follows:

SYS1.VTAMLST(BBIAPPLB)

```

                VBUILD  TYPE=APPL
$BBIB          APPL    AUTH= ( ACQ , NOTSO , VPACE )

```

SYS1.VTAMLST(BBIXSB)

```

                VBUILD  TYPE=CDRSC
$BBIA          CDRSC   CDRM=cdrmname , ISTATUS=ACTIVE

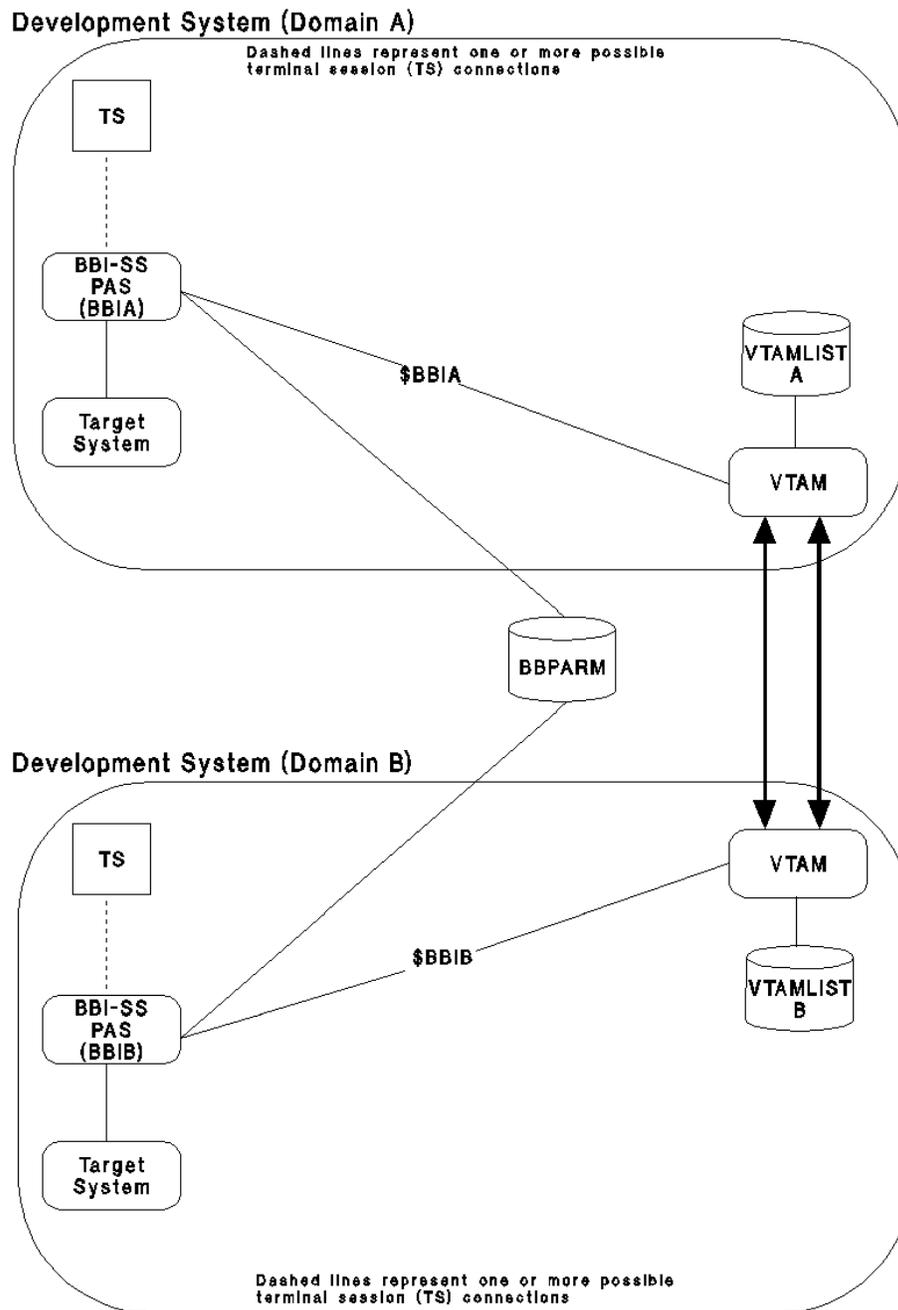
```

where:

\$BBIA Is the cross-domain node for the BBI-SS PAS used in the VTAM application node definition in Domain A.

cdrmname Is the name of the CDRM major node defined for Domain A.

Figure 5-7 BBI-SS PAS to BBI-SS PAS Cross-Domain Communication



Define a CAS to CAS Communication

This section applies only to the following MAINVIEW product groups:

- Group 2
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

Skip this section if:

- All the CASs are running in the same sysplex.

CASs communicate through the XCF link as long as they all have the same XCF group name. The default is BBGROUP.
- The current CAS does not need to communicate with another CAS on a different sysplex through a VTAM link.

See Figure 5-4 on page 5-74.

To create the VTAM definitions necessary for cross-system communication, repeat the following on each system:

- Step 1** Define a VTAM major node for the CAS. See “Define a VTAM Major Node for the CAS” on page 5-87.
- Step 2** Depending on the limitations of the NCPs in your network (if you have them), you may need to create a mode table entry to specify a smaller message size than the default. See “Define a Default VTAM Mode Table Entry” on page 5-89.
- Step 3** Define the CAS(s) you want to communicate with as a CDRM definition. See “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91.
- Step 4** Add the VTAM major node name and CDRM definition(s) to the VTAM configuration list. See “Update the VTAM Configuration List” on page 5-92.
- Step 5** Activate the major node. See “Activate the VTAM Major Nodes” on page 5-93.

Define a VTAM Major Node for the CAS

To define a VTAM major node for the CAS:

Step 1 Create a new member in SYS1.VTAMLST. The member name you use becomes the major node name for the CAS.

Step 2 Add the following APPL statement:

```
VBUILD TYPE=APPL
name APPL AUTH=(ACQ), PARSESS=YES, MODETAB=modetab,
DLOGMOD=BBCEDEF
```

where:

name Is the minor node name for this CAS. This is the name by which VTAM identifies the CAS on the MVS system. The minor node name cannot exceed 8 characters.

In a shared DASD environment, you can define one SYS1.VTAMLST member that contains an APPL statement for each CAS; however, only one VBUILD statement should be defined and each name value you define to an APPL statement must be unique throughout your sysplex or multisystem environment.

modetab Is the name of the mode table that governs LU to LU conversations.

BBCEDEF Is the name of the mode table entry that governs LU to LU conversations

Note: You may need to create a default entry for your MAINVIEW product; otherwise, the VTAM default buffer size for messages could be too small, thus resulting in message truncation. See “Define a Default VTAM Mode Table Entry” on page 5-89 for more information.

Example 1:

To define a separate SYS1.VTAMLST member for each CAS on SYSA, SYSB, and SYSC:

- Step 1** Create a member named ACASA on SYSA that contains this APPL statement:

```
VBUILD TYPE=APPL
CASA  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
```

Once the ACASA definition is activated (see “Activate the VTAM Major Nodes” on page 5-93), the CAS on SYSA is known to VTAM as CASA.

- Step 2** Create a member named ACASB on SYSB that contains this APPL statement:

```
VBUILD TYPE=APPL
CASB  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
```

- Step 3** And on SYSC, create a member named ACASC that contains this APPL statement:

```
VBUILD TYPE=APPL
CASC  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
```

Example 2:

To define one SYS1.VTAMLST member that contains an APPL statement for each CAS in a shared DASD environment, create a single major node member and add the following APPL statements:

```
VBUILD TYPE=APPL
CASA  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
CASB  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
CASC  APPL AUTH=( ACQ ) , PARSESS=YES , MODETAB=BBMODET ,
DLOGMOD=BBCDEF
```

Define a Default VTAM Mode Table Entry

You should create a mode table with the defaults shown in on page 5-90. If you have an entry with these defaults, define the entry as the *modetab* value in “Define a VTAM Major Node for the CAS” on page 5-87.

A default entry allows you to control the RUSIZES (request unit size) parameter that VTAM uses by default to send MAINVIEW application messages to NCPs in your network. If you do not create a default entry, VTAM could select a RUSIZES number that is too small for MAINVIEW product messages, which could result in message truncation.

To create a default mode table entry:

Step 1 Define an entry using the following example:

Figure 5-8 Default Mode Table Entry for the RUSIZES Parameter

```

TITLE 'modetab - MODE Table - Entries'
modetab MODETAB ,
MODEENT LOGMODE=BBCDEF,                                X
      FMPROF=X'13',                                     X
      TSPROF=X'07',                                     X
      PRIPROT=X'B0',                                    X
      SECPROT=X'B0',                                    X
      COMPROT=X'50B1',                                  X
      SSNDPAC=X'00',                                    X
      SRCVPAC=X'00',                                    X
      RUSIZES=X'F8F8',                                  X
      PSNDPAC=X'00',                                    X
      PSERVIC=X'060200000000000000000002300',         X
      ENCR=X'00'
MODEEND ,
END ,

```

where *modetab* is a mode table name that you supply. This must be the same value you defined as the *modetab* value in “Define a VTAM Major Node for the CAS” on page 5-87.

Step 2 Use the sample JCL shown in Figure 5-9 on page 5-91 to assemble the mode table source and link-edit it into SYS1.VTAMLIB on all systems where cross-system communication is to be enabled.

Consider the following:

- You can choose any name you want for the load module. The name you choose becomes the actual name of the mode table.
- You must have access to the macro library used to assemble VTAM applications before you can assemble the mode table source.

- Your system may have a user version of the VTAM definition library concatenated to the SYS1.VTAMLIB library or may use a different library altogether.

Figure 5-9 JCL for Assembling a Mode Table Entry

```
//ASM      EXEC  PGM=ASMBLR, PARM=' OBJECT, NODECK '
//SYSLIB   DD   DSN=SYS1.SISTMAC1, DISP=SHR
//         DD   DSN=SYS1.MACLIB, DISP=SHR
//SYSUT1   DD   DSN=&&SYSUT1, UNIT=VIO, SPACE=(1700, (600, 100))
//SYSTEM   DD   SYSOUT=*
//SYSPRINT DD   SYSOUT=*
//SYSLIN   DD   DSN=&&OBJSET, UNIT=VIO, SPACE=(80, (200, 50)),
//         DD   DISP=(MOD, PASS)
//SYSIN    DD   DSN=dataset_name(member_name), DISP=SHR
// *
//LINK     EXEC  PGM=IEWL, PARM=(XREF, LET, LIST, NCAL, REUS),
//         DD   COND=(8, LT, ASM)
//SYSLIN   DD   DSN=&&OBJSET, DISP=(OLD, DELETE)
//         DD   DDNAME=SYSIN
//SYSLMOD  DD   DSN=SYS1.VTAMLIB(member_name), DISP=SHR
//SYSUT1   DD   DSN=&&SYSUT1, UNIT=VIO,
//         DD   SPACE=(1024, (50, 20))
//SYSPRINT DD   SYSOUT=*
// *
```

If you have a shared DASD environment where all systems share the same SYS1.VTAMLST, you need to do this step only once.

Create CAS Cross-Domain Resource Manager Definitions

To define a CDRM definition for each cross-system CAS you want the current CAS to communicate with:

- Step 1** Create a new member in SYS1.VTAMLST.
- Step 2** Define the cross-system CAS(s) you want to communicate using this CDRSC statement:

```
VBUILD TYPE=CDRSC
name    CDRSC CDRM=cdrm
```

where:

name Is the minor node name for each cross-system CAS (defined as name in “Define a VTAM Major Node for the CAS” on page 5-87).

cdrm Is the VTAM name for the remote MVS system.

- Step 3** Add one CDRSC statement for each remote system you want the current CAS to communicate with.

You should have a naming convention for each CDRM definition member.

Example: If you define three CASs to VTAM as described in “Define a VTAM Major Node for the CAS” on page 5-87 with IDs of SYSA, SYSB, and SYSC, the CDRM name of each MVS system is VTAMA, VTAMB, and VTAMC.

On SYSA: To enable communication between these CASs, a member called CCASA is created in SYSA's SYS1.VTAMLST. CCASA contains these CDRSC statements to define both SYSB and SYSC as cross-domain resources:

```
                VBUILD TYPE=CDRSC
CASB           CDRSC CDRM=VTAMB
CASB           CDRSC CDRM=VTAMC
```

VTAMB and VTAMC are the CDRM names assigned to SYSB and SYSC.

On SYSB: A member called CCASB is created in SYSB's SYS1.VTAMLST. SYSA and SYSC are defined in this member as cross-domain resources with these CDRSC statements:

```
                VBUILD TYPE=CDRSC
CASA           CDRSC CDRM=VTAMA
CASC           CDRSC CDRM=VTAMC
```

On SYSC: A member called CCASC is created in SYSC's SYS1.VTAMLST. SYSA and SYSB are defined in this member as cross-domain resources with these CDRSC statements:

```
                VBUILD TYPE=CDRSC
CASA           CDRSC CDRM=VTAMA
CASB           CDRSC CDRM=VTAMB
```

Update the VTAM Configuration List

Add the new SYS1.VTAMLST members created in “Define a VTAM Major Node for the CAS” on page 5-87 and “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91 to the VTAM configuration list in SYS1.VTAMLST(ATCCON $_{xx}$). ATCCON $_{xx}$ lists all the major nodes that VTAM activates upon initialization. To find ATCCON $_{xx}$, look in the VTAM configuration's start list, SYS1.VTAMLST(ATCSTR $_{xx}$). :lq.

Example: You created:

- SYS1.VTAMLST major node members named ACASA, ACASB, and ACASC for the SYSA CAS, SYSB CAS, and SYSC CAS, respectively, as described in “Define a VTAM Major Node for the CAS” on page 5-87.
- SYS1.VTAMLST CDRM definition members named CCASA, CCASB, and CCASC for each of the respective systems as described in “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91.

To automatically activate the ACASA major node and CCASA CDRM definition at VTAM startup on SYSA, you must add them to SYS1.VTAMLST(ATCCON xx), SYSA's VTAM configuration list member, with these statements:

```
ACASA , X
CCASA , X
```

Note: The X at the far right represents a non-blank continuation character in column 72.

Next, the major nodes created for SYSB, ACASB and CCASB are added to SYSB's configuration list:

```
ACASB , X
CCASB , X
```

And likewise on SYSC:

```
ACASC , X
CCASC , X
```

Note: If all three systems share a SYS1.VTAMLST, you need update only one VTAM configuration list with all three major node IDs and all three CDRM definition names.

Activate the VTAM Major Nodes

To activate the definitions created in “Define a VTAM Major Node for the CAS” on page 5-87 and “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91, you can do one of the following:

- Restart VTAM on each system
- Manually activate the major node definitions

To manually activate the major node definitions for each VTAM configuration list that was updated in “Update the VTAM Configuration List” on page 5-92:

- Step 1** Use the following VTAM command on the respective VTAM to activate new VTAM major nodes:

```
VARY NET,ACT, ID=nodename
```

where *nodename* is the name of the major node member created in “Define a VTAM Major Node for the CAS” on page 5-87.

- Step 2** Verify that the definition is active by using the following VTAM command:

```
D NET, ID=nodename, E
```

- Step 3** Use the following VTAM command on the respective VTAM to activate new CDRM definitions:

```
VARY NET,ACT, ID=cdrmname
```

where *cdrmname* is the name of the CDRM definition member created in “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91.

- Step 4** Verify that the definition is active by using the following VTAM command:

```
D NET, ID=cdrmname, E
```

Customize An XCF Group Name

Skip this section unless one of the following applies to you:

- You cannot use the default XCF group name of BBGROUP.
- You are running multiple CASs on the same system.

MAINVIEW products use a default XCF group name of BBGROUP and each CAS in the group uses a member name that is the same as the CAS name. Both of these values are defined when you create a CAS definition for each CAS in the sysplex.

If you cannot use the default BBGROUP name:

- Step 1** Start the CAS.
- Step 2** Define the same group name in the XCF Group Name field on the ADD CAS SYSTEM DEFINITION or CHANGE CAS SYSTEM DEFINITION panels for all CASs that you want to join an XCF group.
- Step 3** Issue the SAVE command for each CAS definition you add or change using the CASDEF view.

Step 4 Exit from the MAINVIEW window.

Step 5 Stop the CAS:

```
P casname
```

Step 6 Restart the CAS:

```
S casname
```

The *MAINVIEW Administration Guide* provides more information about starting and stopping a CAS.

Define CAS Definitions for the First Time

Once your cross-systems communication environment is established, use the CASDEF view in Plex Manager to define additional CAS definitions, as described below. To maintain existing CAS definitions, see the *MAINVIEW Administration Guide*.

To enable cross-systems communication between CASs, you should follow the instructions below on the local system first, then on each of the systems you want to communicate with:

Step 1 Start the CAS as follows:

```
S casname
```

For more information about starting a CAS, see the *MAINVIEW Administration Guide*.

Step 2 Start the TS (see the *MAINVIEW Administration Guide*) to display the MAINVIEW Selection Menu and select Option 1 for Plex Manager.

The PLEXOVER view is displayed.

Step 3 Enter CASDEF on the COMMAND line and press ENTER to display the CASDEF view.

Note: When you display CASDEF on a system for the very first time and there are no existing definitions, information for a default definition is shown with the following information in the Description field:

```
Use CASDEF to update
```

```
and the VTAM Appl Name field has a value of *NONE*.
```

Important! If you are migrating from a previous product release, these definitions are listed when you display CASDEF. The same description field value appears for all your definitions.

Step 4 Enter EDIT on the COMMAND line to obtain an edit lock.

Step 5 For the local CAS, and then for each cross-system CAS:

Enter ADD *casname* on the COMMAND line to display Figure 5-10 on page 5-96, where *casname* is the name of the MVS system, or any unique name.

Note: When you enter the ADD command and specify a *casname*, you are defining the name of the CAS. You should name each CAS the same as the MVS system name, since only one CAS per MVS image should run, except in trial or test situations where a production CAS is already running on the same system.

Figure 5-10 ADD CAS SYSTEM DEFINITION Panel

```

----- ADD CAS SYSTEM DEFINITION -----
COMMAND ==>

CAS System Name   ==> SYSA      (Recommended same as MVS System Name)
Description       ==>

System Identification Information:
MVS System Name   ==> SYSA      SMF ID           ==> *
SysPlex Name      ==> *        Subsystem ID      ==> *

System Communication Information:
VTAM Appl Name    ==>
XCF Group Name    ==> BBGROUP

Parmlib Suffix Information:
SepDef Suffix     ==> 00        TgtDef Suffix     ==> 00
SecDef Suffix     ==> 00        ConDef Suffix     ==> 00
Common SerDef     ==> 00

Enter END to add the CAS System Definition.
```

The ADD CAS SYSTEM DEFINITION panel allows you to systematically define unique cross-system communication parameters to each CAS.

Step 6 Press your help key for information about each field.

CAS system name and description are required. The other fields provide:

- System identification information

Information provided in these fields uniquely identify each CAS.

- System communication information

Information provided in these fields set up cross system communication for this CAS.

- Parameter library information

Information provided in these fields identify other parameter library definition members created with ADMIN or SECURITY views. This is used primarily in a shared parameter library environment. Keep the default 00 suffixes in each field unless you created specific parameter definitions you want applied to this CAS.

Step 7 When you are finished with this CAS definition, enter END to return to the CASDEF view.

Step 8 Enter SAVE on the COMMAND line on the COMMAND line to save the information in the BBPARM CAS definition member.

Step 9 Repeat Step 5 through Step 8 for each cross-system CAS you want to establish communications with.

Step 10 Stop the CAS (`P casname`) and then start it (`S casname`) to initialize the CAS with new cross-system CAS definitions.

For more information about starting and stopping the CAS, see the *MAINVIEW Administration Guide*

Step 11 The definitions on the local CAS are now complete; however, the remote CAS systems still need information about themselves and the CASs remote to them.

If each CAS has a separate and unique BBPARM library that it does not share with any other CAS, do the following:

11.A Log off the current system.

11.B Log on to each remote system, in turn.

11.C Repeat Step 1 through Step 10.

If the CASs are operating in a shared DASD, shared BBPARM environment, do the following:

11.A Log off the current system.

11.B Log on to each remote system, in turn.

- 11.C** Repeat Step 1 on page 5-95 through Step 5 on page 5-96.
- 11.D** Enter **INStall casname** on the `COMMAND` line to update the cross-system communication parameters for the local CAS.
- 11.E** Stop the CAS (`P casname`) and then start it (`S casname`) to initialize the CAS with the new cross-system CAS definitions.

For more information about starting and stopping a CAS, see the *MAINVIEW Administration Guide*.

Troubleshooting CAS Cross-System Communication

If you have problems activating cross-system communication, see the following table:

If this occurs	Do this
Message BBCSJ015W appears in the SYSLOG with a sense code of 800A0000.	See “Create a Mode Table Entry to Correct a Sense Code of 800A0000.” on page 5-98.
Message BBCSJ008W appears in the SYSLOG with a return code of 5A.	Ensure that you activated the APPL IDs for the CASSs. See “Activate the VTAM Major Nodes” on page 5-93 for assistance.
Message BBCSJ015W appears in the SYSLOG with a sense code of 0857xxxx	Ensure that your cross-domain links are set up correctly. See “Create CAS Cross-Domain Resource Manager Definitions” on page 5-91 for assistance.

Create a Mode Table Entry to Correct a Sense Code of 800A0000.

As specified in the mode table entry created in “Define a Default VTAM Mode Table Entry” on page 5-89, the default RU (request unit) size for MAINVIEW is 3840 bytes. The mode table contains the default parameters that govern LU to LU conversations adhering to the LU 6.2 protocol.

If you receive a message in the SYSLOG that contains a sense code of 800A0000, at least one of the NCPs (Network Channel Programs) in your network has a message buffer smaller than the MAINVIEW default buffer of 3840 bytes. As a result, some of the data traveling to that system is truncated. To prevent this, you must create an entry in the mode table to accommodate the NCP's buffer size.

Do the following instructions on all systems connected to the NCP:

Step 1 Add the following entry to the mode table created in “Define a Default VTAM Mode Table Entry” on page 5-89:

Figure 5-11 VTAM Mode Table Entry for NCP

```

TITLE 'BMODEA - MODE Table - Entries'
BMODEA  MODETAB ,
MODEENT LOGMODE=BBCDEF,                                X
        FMPROF=X'13',                                    X
        TSPROF=X'07',                                    X
        PRIPROT=X'B0',                                    X
        SECPROT=X'B0',                                    X
        COMPROT=X'50B1',                                  X
        SSNDPAC=X'00',                                    X
        SRCVPAC=X'00',                                    X
        RUSIZES=X'F8F8',                                  X
        PSNDPAC=X'00',                                    X
        PSERVIC=X'060200000000000000002300',            X
        ENCR=X'00'
MODEEND ,
END ,

MODEENT LOGMODE=logmode,                                X
        FMPROF=X'13',                                    X
        TSPROF=X'07',                                    X
        PRIPROT=X'B0',                                    X
        SECPROT=X'B0',                                    X
        COMPROT=X'50B1',                                  X
        SSNDPAC=X'00',                                    X
        SRCVPAC=X'00',                                    X
        RUSIZES=X'yyyy', (see below)                    X
        PSNDPAC=X'00',                                    X
        PSERVIC=X'060200000000000000002300',            X
        ENCR=X'00'
MODEEND ,
END ,

```

where *logmode* is the name of the entry you are adding. (The X at the far right represents a non-blank continuation character in column 72.)

For the RUSIZES parameter, specify a number as close to 3840 bytes as possible:

- Check the MAXDATA parameter in the NCPGEN for the NCP used to communicate with the failing system.
- To accommodate as much data as possible, use the following table to find the number closest to this MAXDATA value, then specify the hexadecimal representation of that number with the mode table's RUSIZES parameter. The number you choose must be at least 15 bytes smaller than the MAXDATA value.

Example: Suppose the MAXDATA value for an NCP is 3335. Looking at the following table, the number closest to 3335 is 3328. However, 3328 is not 15 bytes less than 3335; therefore, the correct number to specify is 3072.

To determine 3072's hexadecimal representation, find the number's value in the A column, which is 12 (C), then its value in the B column, which is 8. Specify C8C8 on the RUSIZES parameter.

You need to repeat the digits, because the RUSIZES parameter contains two bytes. The first byte (AB) reflects the size of the NCP's incoming buffer; the last byte (AB) reflects the size of the outgoing buffer. Usually, these two numbers are the same. '

B	A							
	8	9	A (10)	B (11)	C (12)	D (13)	E (14)	F (15)
5	256	288	320	352	384	416	448	480
6	512	576	640	704	768	832	896	960
7	1024	1152	1280	1408	1536	1664	1792	1920
8	2048	2304	2560	2816	3072	3328	3584	3840
9	4096							

Step 2 Assemble the mode table source and link-edit it into SYS1.VTAMLIB on all CAS systems where cross-system communication is enabled. Use the same instructions and sample JCL in "Define a Default VTAM Mode Table Entry" on page 5-89 for this step.

Step 3 In the member containing the system VTAM definition (created in "Define a VTAM Major Node for the CAS" on page 5-87), add the MODETAB operand to the APPL statement as shown:

```

          VBUILD TYPE=APPL
name     APPL AUTH=(ACQ), PARSESS=YES, MODETAB=module,
DLOGMOD=BBCDEF

```

where:

name Is the name you choose to assign to the MVS image for the CAS.

module Is the name of the load module created in Step 1 on page 5-99 above in these instructions and shown in Figure 5-11 on page 5-99 (BBMODEA in the example).

Step 4 Enter `VARY NET , INACT , ID=name` to deactivate this major node member.

Note: This causes a service interruption for any active TSs.

Step 5 Enter `VARY NET , ACT , ID=name` to reactivate the member and make the MODETAB parameter accessible to VTAM.

Step 36: (Required) Specify Product Initialization within a BBI-SS PAS

This section applies only to the following MAINVIEW product groups:

- Group 1
- Group 3

See Table 4-1 on page 4-2 for a list of products in each group.

BBPARM Member BBISSPxx:

Use this member to enable your product(s) for initialization within the BBI-SS PAS and review other optional parameters that are described later in these instructions.

Step 1 Edit BBISSPxx to specify:

PRODUCT=*prd*

Specify one statement per product with each statement on a separate line; for example:

PRODUCT=CMR
PRODUCT=DMR

where *prd* can be:

AAO	MAINVIEW AutoOPERATOR
CMR	MAINVIEW for CICS
DMR	MAINVIEW for DB2
MVDBC	MAINVIEW for DBCTL
MFP	MAINVIEW FOCAL POINT
MVIMS	MAINVIEW for IMS Online
MVMQS	MAINVIEW for WebSphere MQ (formerly known as MAINVIEW for MQSeries) and MAINVIEW for WebSphere MQ Integrator
MVVP	MAINVIEW VistaPoint

Step 37: (Optional) Enable Automatic Restart Manager Support

This section applies only to MAINVIEW AutoOPERATOR.

Note: The Automatic Restart Manager (ARM) was introduced in MVS/ESA Version 5, Release 2. For ARM support to be available, you must be using MVS/ESA 5.2 or above.

BBPARM Member BBISSPxx:

Use this member to enable support for ARM if you want it to automatically restart BBI-SS PASs that have abended.

Automatic Restart Manager (ARM) provides support for automatically restarting batch jobs and started tasks after unexpected termination. This function is available in either single SYSPLEX or multisystem SYSPLEX using couple data sets.

To enable ARM support, edit a copy of BBPARM member BBISSPxx and specify:

```
ARMPOLCY=[NO|YES|xxxxxxx]
```

where:

NO The BBI-SS PAS will not use the ARM facility.

YES The BBI-SS PAS will use the ARM default restart policy.

The default ARM restart policy is for ARM to restart an abended address space up to 3 times within 5 minutes. If the address space fails to restart after the third attempt, a message is written to the SYSLOG and the address space remains unstarted.

xxxxxxx Specifies a 1- to 8-character name of a user-created policy to be used to control BBI-SS PAS restarts.

For more information about implementing ARM support for MAINVIEW AutoOPERATOR, see the *MAINVIEW AutoOPERATOR Customization Guide*.

Step 38: (Required) Specify Product Option Password Keys

This section applies only to the following products:

- MAINVIEW AutoOPERATOR
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for IMS Offline

Note: This step is not required for MAINVIEW for IMS Online or MAINVIEW for DBCTL, if you use the product authorization utility to activate the products. For information on the product authorization utility, see “Step 6: (Required) Apply Passwords with the Product Authorization Utility” on page 5-9.

BBPARM Member BBKEYS:

Create a BBPARM member called BBKEYS to specify MAINVIEW AutoOPERATOR, MAINVIEW for DBCTL, MAINVIEW for IMS Online, or MAINVIEW for IMS Offline product option password keys.

These products have features and options that require password keys based upon information about your CPU. To obtain password keys:

- Step 1** Determine the internal model and version number of the CPU(s) installed at your site.
- Step 2** Obtain your password keys from BMC Software.

Determining the Internal CPU Model and Version Number:

Choose one of the following:

- To display your existing password keys and the current CPU ID:
 - Enter the following BBI control command from your MAINVIEW terminal session:

```
.D KEYS
```

The command output is written to the BBI-SS PAS journal log.

— Or, enter the following MVS MODIFY command:

```
/F ssid, .D KEYS
```

where *ssid* is the BBI-SS PAS job name or started task ID. The command output is written to the SYSLOG or CONSOLE.

Both commands produce output as follows. The next to the last line identifies the current CPU ID.

```
CF8701I COPY PROTECTION KEY STATUS:
CF8702I PRD CPUID EXP SYS STATUS
CF8703I --- ----- --- --- -----
CF8704I ANV-9021-38-72363-99365-3762 VALID
CF8704I CAO-9021-38-72363-99365-968C VALID
CF8704I IAO-9021-38-72363-99365-D959 VALID
CF8707I MAO-****-**-*****-99365-2151 EXPIRES IN 0043
DAYS
ON 19-JAN-02
CF8705I -----
CF8706I CURRENT CPUID IS 9021-38-C2363
CF8705I -----
CF8708I KEYS/STATUS OBTAINED AT PAS STARTUP ON
07-DEC-01
```

- Extract the CPU information by using the BMC Software SYSINFO utility

1. On the MVS system where these products are to execute, enter:

```
TSO CALL 'hilevel.BBLINK(SYSINFO)'
```

where *hilevel* represents your high-level data set qualifier.

2. Record the information from the SYSINFO utility below.

```
CPU IDENTIFIER: _ _ _ _ _
CPU TYPE: _ _ _ _
CPU MODEL: _ _
```

- Use TSO TEST to display storage on the active system by entering the following from the TSO READY prompt:

```
TEST `SYS1.LINKLIB(IEFBR14)'
```

Press enter, then type:

L 208.%+4 L(12) C

The TEST LIST subcommand displays the following output:

```
aaaaaaaa. XXnYYYYYZZZZ
```

where

aaaaaaaa	Represents hexadecimal characters (not needed for password key)
XX	Is the CPU model version
n	Represents a numeric (not needed for password key)
YYYYY	Is the CPU serial number (CPU ID)
ZZZZ	Is the CPU type

Obtaining Password Keys from BMC Software:

You should have received new password key(s) from BMC Software via e-mail shortly after your product license agreement was processed. If you have problems or questions, contact the Customer Password Response team at 800 841 2031 (USA or Canada) or your local BMC Software office.

Edit BBPARM member BBKEYS (create one if you do not have one) and add the key parameters.

If you receive more than one key, each key must begin on a separate line. During key validation, MAINVIEW attempts to use the most suitable key by validating each key and selecting the one most qualified for your CPU.

Specify:

```
KEY=ppp-cccc-tt-nnnnn-yyddd-xxx[-sss]
```

where:

ppp Is a 3-character product ID; such as:

- ANVMAINVIEW AutoOPERATOR Access NV
- CAOMAINVIEW AutoOPERATOR for CICS
- IAOMAINVIEW AutoOPERATOR for IMS
- MAOMAINVIEW AutoOPERATOR for OS/390

QAOMAINVIEW AutoOPERATOR for MQSeries

TSHMAINVIEW AutoOPERATOR TapeSHARE

IONMAINVIEW for IMS Online

IOFMAINVIEW for IMS Offline

IPXIMSPlex System Manager

IRAIMS Resource Analyzer

IRMIMS Resource Monitor

IWAIMS Workload Analyzer

IWMIMS Workload Monitor

IADIMS Workload Analyzer Extension for DB2

IMDIMS Workload Monitor Extension for DB2

ILMIMS Resource Analyzer Extensions for IRLM and IMS Resource
Monitor Extensions for IRLM

IPFIMS Performance Reporter

IPDIMS Performance Reporter Extension for DB2

ITAIMS Transaction Accountant

ITDIMS Transaction Accountant Extension for DB2

DBCMAINVIEW for DBCTL

Note: To use the MAINVIEW AutoOPERATOR TapeSHARE option (TSH), MAINVIEW AutoOPERATOR for OS/390 must be installed and the MAO key must also be specified. See the *MAINVIEW AutoOPERATOR Customization Guide* for more information.

<i>cccc</i>	Is a 4-character CPU type (for example, 3090).
<i>tt</i>	Is a 2-character CPU internal model group type (for example, 23).
<i>nnnn</i>	Is a 4- to 5-digit CPU serial number in hexadecimal format.
<i>yyddd</i>	Is the Julian expiration date.
<i>xxxx</i>	Is a 4-character authorization key created by BMC Software.

Note: The values for *ppp*, *cccc*, *tt*, or *nnnn* can be a generic * qualifier; however, the password key created by BMC Software must have been generated using this same qualifier.

ssss Is the SMF ID of the system where the product option should be activated.

The use of *ssss* helps prevent an IPL failure because it identifies an invalid product password key and allows for immediate correction. When an invalid key is detected, a WTOR message is issued and the operator can specify another key immediately. If the SMF ID of the target system is not specified, no external warning messages are issued for invalid keys.

Warning! If your CPU serial number changes or an additional CPU is added, BMC Software must receive the new number so that a new password key can be generated.

Step 39: (Required) Add CAS ID to BBI-SS PAS

This section applies only to Group 2 products. See Table 4-1 on page 4-2 for a list of products in each group.

BBPARM Member BBISSPxx:

Use your copy of this member to identify the CAS to the BBI-SS PAS.

The BBI-SS PAS connects to the CAS that is active in the same MVS image.

Edit BBISSPxx to:

- Step 1** Add the `CASID=` statement.
- Step 2** Specify the 4-character subsystem ID of the CAS to which this BBI-SS PAS is to connect.

For example, the default CAS ID in the distributed BBISSPxx is:

```
CASID=BBCS
```

Note: The CAS subsystem ID is defined with the PROC statement in the CAS started task procedure (see “BBILIB member @@YZZ021 contains statements for creating a CAS startup procedure in your system started task procedure library. These statements are described in the following table.” on page A-1).

Step 40: (Optional) Add Extended Function Data Sets to BBI-SS PAS StartUp Procedure

This section applies only to Group 2 products. See Table 4-1 on page 4-2 for a list of products in each group.

BBSAMP Member SSJCL:

Use your copy of this member to add the following data sets to the BBI-SS PAS started task procedure (see “Step 31: (Required) Create a BBI-SS PAS Start Procedure” on page 5-59):

HISTDS <i>nn</i>	Historical data sets
BBACTDEF	Action and view table data set
BBVDEF	Default views data set

Edit SSJCL to:

- Step 1** Add a //HISTDS*nn* DD statement for each historical data set.
- Step 2** Add a //BBACTDEF statement.
- Step 3** Add a //BBVDEF statement.

The example below shows DD statements used to add three historical data sets for a BBI-SS PAS running on MVSA. Separate DD statements initialize the action and view table data set and the default views data set.

```
//HISTDS00 DD DSN=hilevel.MVSA.HISTDS00,DISP=SHR
//HISTDS01 DD DSN=hilevel.MVSA.HISTDS01,DISP=SHR
//HISTDS02 DD DSN=hilevel.MVSA.HISTDS02,DISP=SHR
//BBACTDEF DD DSN=hilevel.BBACTDEF,DISP=SHR
//BBVDEF DD DSN=hilevel.BBVDEF,DISP=SHR
//PARMFILE DD DSN=hilevel.PARMFILE,DISP=SHR
```

After you install historical data sets, use the MAINVIEW DSLIST command to view and verify their status. The DSLIST views shows the name and current status of each data set. DSLIST also shows the beginning and ending date and time of the most recent recording interval. You can allocate another historical data set by issuing the ADD command from DSLIST.

Step 41: (Required) Update TSO Command Processor Table

This section applies to all MAINVIEW products.

The BMC Software programs/commands listed below execute as command processors for some of the options from the MAINVIEW Selection menu. If your site has a security package, such as RACF, CA-ACF2, PCF, or EMTMPW, that defines an Authorized Command Processor Table to restrict command processor execution, the following programs/commands may need to be added to this table:

BALCMMSG
BBM3API
BBM9TC21
BBM9TC22
BBM9TC24
BBVJSETP
EMTMPW
SMLOAD
TSLOAD

To add these programs to the command table, see the documentation for the security program used at your site. If these programs are not in the command table, a COMMAND NOT FOUND message is issued when an option that uses them is selected from the MAINVIEW Selection Menu.

Authorizing the TS as a TSO Command Processor:

(Applies to Group 1 and Group 3 products only) If you are invoking the MAINVIEW TS (terminal session) in an ISPF environment or you are calling TSLOAD as a command processor, you must verify that your security environment allows command processors to execute under ISPF. If not, the message SMLOAD NOT FOUND is issued. Check the following, consulting the person responsible in each respective area:

- Step 1** Does the ISPF TSO Command Table (ISPTCM) allow TSLOAD and SMLOAD to be issued as a TSO command processor? If not, see the IBM publication *Interactive System Productivity (ISPF) and ISPF/Program Development Facility for MVS Installation and Customization* and add TSLOAD and SMLOAD to the ISPTCM table.

Note: The ISPTCM entry for TSLOAD and SMLOAD must specify a FLAG parameter with the X'40' and X'02' bits set on (that is, ISPTCM must be set to X'42').

- Step 2** If the Program Control Facility (PCF) is installed on your system, does it allow the SMLOAD and TSLOAD commands to be issued as a TSO command processor? If not, either add SMLOAD and TSLOAD to the PCF-authorized Program Table or set the user's authorization level to zero.
- Step 3** If you have RACF or other appropriate security installed, does it allow the SMLOAD command to be issued as a TSO command processor? If not, change the ACF2 command list to do so.
- Step 4** If you have checked all of the above and still get the SMLOAD NOT FOUND message, turn off all security checking and run the TS. If it runs, then the problem is with one of the site's security programs. If the problem persists, contact BMC Software Customer Support.

Step 42: (Optional) Specify Product Initialization within the MVS PAS

This section applies only to the following MAINVIEW products:

- CMF MONITOR
- MAINVIEW for OS/390
- MAINVIEW for UNIX System Services
- MAINVIEW SYSPROG Services

BBPARM Member BBATSP00:

Use this member to control the initialization of the products listed above within the MVS PAS. By default, all of these products are initialized, however, you can suppress the initialization of one or more products using BBATSP00.

To copy BBATSP00 to a UBBPARM library:

- Step 1** Copy *hilevel*.BBSAMP member BBAMVCPY to a private JCL library, where *hilevel* is the high level qualifier used for MAINVIEW for UNIX System Services data sets at your site. BBAMVCPY contains JCL to execute IEBCOPY. It will copy member BBATSP00 to a UBBPARM library.
- Step 2** Customize the JCL in BBAMVCPY by following the instructions at the top of the member.
- Step 3** Submit the JCL.
- Step 4** Review the job output to verify that the IEBCOPY job was successful.

To enable or disable the initialization of a product:

- Step 1** BBATSP00 uses the following syntax to enable or disable a product:

```
<product>
  <name>      ppppp      </name>
  <server>    ssssssss   </server>
  <target>    tttttttt   </target>
  <enable/>   or <disable/>
</product>
```

where:

ppppp Is one of the following product abbreviations:

MV390 MAINVIEW for OS/390
MVUSS MAINVIEW for UNIX System Services
MVSPS MAINVIEW SYSPROG Services
CMF CMF MONITOR

Note: When enabling MAINVIEW for OS/390, ensure that MAINVIEW SYSPROG Services is also enabled to activate the SYSPROG Services component of MAINVIEW for OS/390.

ssssssss Is the 1- to 8-character job name of the MVS PAS.

tttttttt Is the 1- to 8-character name of the context for the MVS PAS.

Step 2 To disable a product, change `<enable/>` to `<disable/>`.

Step 3 Save the member and restart the PAS.

Appendix A Library Member for CAS

This section describes the BBLIB library member you can use to create the CAS startup procedure.

BBLIB member @@YZZ021 contains statements for creating a CAS startup procedure in your system started task procedure library. These statements are described in the following table.

Table A-1 CAS Started Task Procedure Statements

Statement	Description
// <i>procname</i> PROC	See Table A-2 on page A-2 for an explanation of each PROC statement parameter. You can override these parameters with the START command for the CAS, as described in the <i>MAINVIEW Administration Guide</i> .
//CAS EXEC	The program named on the EXEC statement, BBM9ZA00, processes EXEC parameters and performs basic initialization tasks. The PARM parameter allows you to override default parameter values specified with the PROC statement. TIME=1440 defines unlimited processing time for the CAS. REGION=4096K specifies the size of the private region required by the CAS. Do not define a region smaller than 4096.
//STEPLIB DD	Defines the BBLINK load library, which contains load modules for all MAINVIEW products. The CAS requires some of these load modules; each product address space requires others. This statement should be deleted if BBLINK is added to your system link list.

Table A-1 CAS Started Task Procedure Statements (continued)

Statement	Description
//BBSECURE DD	<p>Defines the data set that contains security resource definitions. These definitions are used by the CAS and PAS to determine if security checking should occur for system, product, view, and/or action access. The same data set name must be defined for the CAS started task procedure BBSECURE statement and the PAS started task procedure BBSECURE statement.</p> <p>Defining a BBSECURE data set is only part of setting up security with your external security manager (ESM). For more information, see <i>Implementing Security for MAINVIEW Products</i>.</p>
//BBACTDEF DD	<p>Defines the data set that contains distributed action and view tables. These tables help the CAS determine valid view names and actions. It also contains default security parameter and resource definition files.</p> <p>The CAS accesses this data set in response to window commands and actions associated with the Plex Manager views.</p>
//BBVEDF DD	<p>Defines the library that contains distributed views.</p>
//BBIPARM DD	<p>Defines and concatenates MAINVIEW product parameter libraries.</p> <p>If you establish cross-system communication or create target or SSI contexts, all definitions are stored in <i>hilevel.UBBPARM</i> created by AutoCustomization.</p> <p><i>hilevel.BBPARM</i> contains distributed parameters.</p>

The following table describes the parameters used for the PROC statement in the CAS started task procedure (see Table A-1 on page A-1).

Table A-2 CAS PROC Statement Parameters

Parameter	Description
procname	<p>Required; the name of the startup procedure for the CAS; for example, BBMCAS as created by AutoCustomization.</p>
SSID= <i>ssid</i>	<p>Required for the CAS startup procedure, but optional for the START command. <i>ssid</i> identifies the CAS subsystem ID to the UAS and PAS. If an SSID is not specified for the START command, the SSID for the PROC statement in the CAS startup procedure is used by default.</p> <p>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.</p> <p>For a PAS to connect to a CAS, the SSID values in both the CAS and PAS startup procedures must be the same.</p>
CAPS={Y N}	<p>Optional; identifies whether the CAS should provide katakana terminal support for console and WTO messages issued from the CAS. The default is N (NO); console and WTO messages appear in mixed case.</p> <p>If you initialize the CAS with <i>CAPS=Y</i>, console and WTO messages are converted to uppercase for katakana terminals.</p>

Table A-2 CAS PROC Statement Parameters (continued)

Parameter	Description
CONVXCF={Y N}	<p>Optional; indicates whether a CAS uses the Cross-System Coupling Facility (XCF) to communicate with other CASs in a sysplex environment. The default is Y (YES), and BMC Software strongly recommends using the default.</p> <p>CONVXCF=Y indicates that XCF is to be used as the primary communication method between CASs. All participating CASs must reside in the same sysplex and XCF group (as identified by the DFLTGRP= parameter). VTAM will be used to communicate with a CAS that is not in the same sysplex.</p> <p>If all the CASs are in the same sysplex and XCF group, CONVXCF=Y allows the CAS to auto-discover and auto-connect to the other CASs through XCF. All the existing VTAM APPLID definitions in the CASDEF entries for individual CASs can be removed.</p> <p>For additional information on XCF support, refer to the <i>MAINVIEW Administration Guide</i>.</p>
COLD={Y N}	<p>Optional; indicates whether CAS startup should be a cold start. The default is N (NO).</p> <p>Specify COLD=Y only upon the request of BMC Software Customer Support in an attempt to clear an error condition. Inappropriate use of COLD=Y may exhaust available MVS linkage indexes, a condition which requires an IPL to restore normal system operation.</p> <p>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, MVS system linkage indexes. In addition, two load modules are retained in common storage.</p> <p>COLD=N reuses those control blocks and load modules rather than building new ones. COLD=Y causes the linkage indexes, control blocks and load modules to be discarded. The linkage indexes and common storage occupied by those resources are lost until the system is IPLed.</p>
DFLTGRP= <i>name</i>	<p>Optional; identifies the Cross-system Coupling Facility (XCF) group name to be used at CAS start up. The default is BBGROUP.</p> <p>DFLTGRP= can be used during migration from one release of MAINVIEW to another. You can use it to start a test CAS on an MVS system that is already running a production CAS. By specifying different XCF group names, you can run two CASs on the same system.</p>
DUMP={N Y ALL}	<p>Optional; indicates whether system dumps (SDUMPs) are taken when a severe error is detected by a CAS. The default is Y (YES); dumps are taken when the error occurs in privileged code.</p> <p>DUMP=N supresses all dumps within the CAS.</p> <p>DUMP=ALL allows dumps for all errors, whether the error occurs in privileged or problem-state code. You should use ALL only as instructed by BMC Software Customer Support.</p>

Table A-2 CAS PROC Statement Parameters (continued)

Parameter	Description
EMM={Y N}	<p>Optional; indicates whether extended message mode (EMM) is active when starting the CAS. The default is N (NO).</p> <p>EMM messages are a subset of messages controlled by the XDM parameter. You can enable EMM messages by specifying <code>EMM=Y</code> without specifying XDM messages. However, <code>EMM=Y</code> is not recommended for normal CAS operation (only as instructed by BMC Software Customer Support).</p>
SPCF=Y	<p>Required; indicates that the CAS uses the Cross-System Coupling Facility (XCF) to communicate with other CASs in a sysplex environment. SPCF=Y is automatically set at CAS start up because XCF is required to support certain CAS subtasks.</p> <p>Note: If SPCF=N is specified, the parameter is ignored and the following message is displayed:</p> <p>BBMZA101I SPCF=Y is required for MAINVIEW Infrastructure (MVI) version 4.0.0 and later</p>
XDM={Y N}	<p>Optional; indicates whether the CAS operates in extended diagnostic mode (XDM), which produces additional messages and suspends some CAS error recovery processing. The default is N (NO).</p> <p>Warning: Only specify <code>XDM=Y</code> as instructed by BMC Software Customer Support. The <code>XDM=Y</code> parameter disables certain error recovery mechanisms that could severely impact your CAS subsystem.</p>

Appendix B Library Member for MVS PAS

This section describes the BBLIB library member you can use to create the MVS PAS startup procedure.

BBILIB member BBMPAS contains statements for creating a PAS startup procedure in your system started task procedure library. These statements are described in the following table.

Table B-1 MVS PAS Started Task Procedure Statements

JCL Statement	Description
// <i>procname</i> PROC	See Table B-2 on page B-4 for an explanation of each PROC statement parameter. You can override these parameters with the START command for the PAS, as described in the <i>MAINVIEW Administration Guide</i> .
//PAS EXEC	The program named on the EXEC statement, BBM9DA00, processes EXEC parameters and performs basic initialization tasks. The PARM parameter allows you to override default parameter values specified with the PROC statement. REGION parameter specifies the size of the private region required by the PAS. TIME=1440 defines unlimited processing time for the PAS.
//STEPLIB DD	Defines the BBLINK load library, which contains load modules for all MAINVIEW products. The CAS requires some of these load modules; each product address space requires others. This statement should be deleted if BBLINK is added to your system link list.

Table B-1 MVS PAS Started Task Procedure Statements (continued)

JCL Statement	Description
//CMFCPMxx DD	<p>(CMF MONITOR only.) Specifies one or more data sets where the Extractor records data collected by the CPM monitor. Do not define this statement if you are recording CPM data to SMF or if you are specifying the data set names on the REPORT Extractor control statement.</p> <p>To manually customize the CMFCPMxx DD statements to point to your CPM data sets, remove the DUMMY parameter and specify the following for each data set you want defined during Extractor initialization:</p> <pre>//CMFCPMXX DD DISP=SHR,DSN=&BAVSHLQ. .&SYSID. .CMFOUTXX</pre> <p>where xx is a unique value for the allocated data set.</p> <p>The //CMFCPMxx DD statements can be defined to write simultaneously to the same data sets as the //CMFIPMxx DD statements. During AutoCustomization, these statements are dynamically defined, if required.</p> <p>Specifying alternate data sets is optional; however, alternate data set support is provided only if an alternate data is specified.</p> <p>See the <i>CMF MONITOR Customization Guide</i> for more information about the CPM monitor, allocating CPM data sets, and alternate data set support; see the <i>CMF Monitor Batch User Guide and Reference</i> for more information about the REPORT Extractor control statement.</p>
//CMFIPMxx DD	<p>(CMF MONITOR only.) Specifies one or more data sets where the Extractor records data collected by the IPM monitor. Do not define this statement if you are recording IPM data to SMF or if you are specifying the data set names on the REPORT Extractor control statement.</p> <p>To manually customize the CMFIPMxx DD statements to point to your IPM data sets, remove the DUMMY parameter and specify the following for each data set you want defined during Extractor initialization:</p> <pre>//CMFIPMXX DD DISP=SHR,DSN=&BAVSHLQ. .&SYSID. .CMFOUTXX</pre> <p>where xx is a unique value for the allocated data set.</p> <p>The //CMFIPMxx DD statements can be defined to write simultaneously to the same data sets as the //CMFCPMxx DD statements. During AutoCustomization, these statements are dynamically defined.</p> <p>Specifying alternate data sets is optional; however, alternate data set support is provided only if an alternate data is specified.</p> <p>See the <i>CMF MONITOR Customization Guide</i> for more information about the IPM monitor, allocating IPM data sets, and alternate data set support; see the <i>CMF Monitor Batch User Guide and Reference</i> for more information about the REPORT Extractor control statement.</p>

Table B-1 MVS PAS Started Task Procedure Statements (continued)

JCL Statement	Description
<p>//CMFCDSxx DD or //CMFIDSxx DD</p>	<p>(CMF MONITOR only.) Specifies one or more data sets for DSO data from the IPM and CPM modes. If defined, all CMF 240 records produced by the HEADMOVE control statement are directed to these data sets.</p> <p>See the <i>DSO User Guide and Reference</i> for more information about the HEADMOVE control statement and allocating DSO data sets.</p> <p>These DD statements function the same way as the //CMFCPMxx DD and //CMFIPMxx DD statements.</p> <p>Do not define this statement if you are recording DSO data to SMF or if you are specifying the data set names on the HEADMOVE Extractor control statement.</p>
<p>//CMFDUMP DD</p>	<p>(CMF MONITOR only.) Defines a print data set for snapshot dumps. When a program fails in CMF MONITOR Online, the abend is intercepted by an ESTAE routine and a formatted dump of the failing component is printed.</p>
<p>//CMFLOG DD</p>	<p>(CMF MONITOR only.) Defines an optional file that can be used to direct the CMF Control Card Log to an alternative data set.</p> <p>The data set for the Control Card Log is dynamically allocated and deallocated as needed when no //CMFLOG DD statement is present. The CMFLOG data set allows for immediate inspection of messages, regardless of how long an Extractor job runs. Dynamic allocation is to the MSGCLASS= specified on the JOB statement for batch jobs or CLASS A for started tasks.</p>
<p>//CMFMSG DD</p>	<p>(CMF MONITOR only.) Defines a print data set that contains non-initialization messages for the system. The //CMFMSG DD data set is dynamically allocated to SYSOUT class A when no DD statement is present.</p> <p>To print this data set, issue a MSGFREE command by using the MVS MODIFY command. The data set is dynamically deallocated and printed, and a new allocation occurs immediately.</p>
<p>//CMFXDSLGD DD</p>	<p>(CMF MONITOR only.) Defines a print data set that contains XDS initialization messages and control cards.</p> <p>If this DD statement is not present, XDS messages are issued as WTO.</p>
<p>//PARMLIB DD</p>	<p>(CMF MONITOR only.) Defines a partitioned data set that contains control statement members the Extractor reads. The PARMLIB data set must be a fixed-block data set with a logical record length of 80, such as <i>hilevel.UBBPARM</i>.</p>

Table B-1 MVS PAS Started Task Procedure Statements (continued)

JCL Statement	Description
//BBSECURE DD	<p>Defines the data set that contains security resource definitions. These definitions are used by the CAS and PAS to determine if security checking should occur for system, product, view, and/or action access. The same data set name must be defined for the CAS started task procedure BBSECURE statement and the PAS started task procedure BBSECURE statement.</p> <p>Defining a BBSECURE data set is only one part of setting up security to function with your external security manager (ESM). For more information about customizing security, see <i>Implementing Security for MAINVIEW Products</i>.</p>
//BBVDEF DD	Defines the library that contains distributed views.
//BBACTDEF DD	Defines the data set that contains distributed action and view tables.
//USSDUMP DD	(MAINVIEW for UNIX System Services only.) Used by the OEDGTRC trace function for problem diagnosis at the request of BMC Software Customer Support.
//PARMFILE DD	<p>Defines a VSAM data set that contains:</p> <ul style="list-style-type: none"> Names of historical data sets for the DSLIST view. User-defined workload definitions created through the MAINVIEW for OS/390 WKLIST view. <p>This data set is allocated and initially loaded during the customization process.</p>
//HISTDS00 DD //HISTDS01 DD //HISTDS02 DD	Defines up to 100 VSAM data sets that comprise the historical database.

The following table describes the parameters for the PROC statement used in the MVS started task procedure (see Table B-1 on page B-1).

Table B-2 MVS PAS PROC Statement Parameters

PROC Parameter	Description
procname	Required; the name of the startup procedure for the PAS; for example, MVSPAS.
SYSID= <i>sysid</i>	Required; specifies the SMF system ID for your system.
XDM={Y N}	<p>Optional; specifies whether the MVS PAS should execute in extended diagnostic mode (XDM), which produces additional messages and suspends some PAS error recovery processing. The default is N (NO).</p> <p>Warning: Only specify XDM=Y as instructed by BMC Software Customer Support. The XDM=Y parameter disables certain error recovery mechanisms that could severely impact your system.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
CXEN={Y N}	<p>Optional (CMF MONITOR 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.</p> <p>You can run more than one CMF Extractor on your system at a time. However, only one Extractor can sample workload data (MVS 4.3 and below) or enqueue data.</p> <p>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.</p> <p>You should not run multiple Extractors because of associated overhead, unless it is required in trial situations to evaluate MAINVIEW for OS/390. CXEN=N should be defined to a trial Extractor, if necessary.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
DC={CPM IPM <u>START</u> STOP}	<p>Optional (CMF MONITOR and MAINVIEW for OS/390 only). affects the samplers associated with the PGDDLAY and CFDATA Extractor control statement and the MVS PAS data collectors; it has the following attributes:</p> <p>START The default; MVS PAS data collectors initialize at startup under CPM monitoring mode. When the data collectors are initialized, access to CMF MONITOR Online and/or MAINVIEW for OS/390 is provided. The CMF samplers associated with the PGDDLAY and CFDATA Extractor control statements are enabled for sampling.</p> <p>STOP Data collectors do not initialize when the MVS PAS initializes. You cannot access CMF MONITOR Online and/or MAINVIEW for OS/390. CMF's PGDDLAY and/or CFDATA Extractor control statements do not sample data because both gather information from the MVS PAS data collectors.</p> <p>CPM MVS PAS data collectors initialize and accept data from the Extractor in CPM mode at startup. Specifying DC=CPM is the same as DC=START.</p> <p>IPM MVS PAS data collectors initialize and accept data coming from the Extractor in IPM mode at startup.</p> <p>If you specify DC=IPM, you must also specify the parameter IPM=xx, where xx is the suffix of the CMFIPMxx control statement member pointed to by the //PARMLIB DD statement.</p> <p>The DC parameter can also be changed by using an MVS MODIFY command, so the data collectors can be started or stopped or the monitoring modes switched without requiring the MVS PAS (and Extractor) to be stopped and then started. There is an additional operand you can specify with the DC= parameter when the MVS MODIFY command is issued:</p> <p>STATUS Status of each data collector is displayed in the job log.</p> <p>Warning: C=STATUS should not be specified at MVS PAS initialization because the data collectors do not initialize. It should be used only when the PAS is running.</p> <p>For more information about how to use the MODIFY command to change MVS PAS operation, see the <i>MAINVIEW Administration Guide</i>.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
CPM={xx 00}	<p>Optional (CMF MONITOR and MAINVIEW for OS/390 only). Specifies the two-character suffix of the CMFCPMxx control statement member in the data set pointed to by the //PARMLIB DD statement.</p> <p>The member pointed to by the CPM parameter must be named CMFCPMxx, where xx is the suffix defined with CPM=. The default is CPM=00.</p> <p>The control statement member contains Extractor REPORT and sampler control statements. This member controls the operation of the CPM mode samplers and defines the resources the Extractor monitors.</p> <p>CPM mode samplers execute either when a MODIFY command is issued or when the CPM parameter is used. Sample control statement members are in <i>hilevel</i>.UBBPARM data set created by AutoCustomization.</p> <p>The CPM mode control statement member the Extractor uses can be changed while the Extractor is active by using the MODIFY command; see the <i>MAINVIEW Administration Guide</i> for more information.</p> <p>Warning: If CPM=STOP is issued, the address space is terminated unless IPM monitoring mode is active.</p>
IPM={xx 00}	<p>Optional (CMF MONITOR and MAINVIEW for OS/390 only). Specifies the two-character suffix of the CMFIPMxx control statement member in the data set pointed to by the //PARMLIB DD statement. The default is IPM=STOP.</p> <p>You should add this parameter to the PAS started task procedure only if you always want to start the IPM monitor when the CPM monitor starts.</p> <p>The member pointed to by the IPM parameter must be named CMFIPMxx, where xx is the suffix defined with IPM=. The default is IPM=00.</p> <p>The control statement member contains the Extractor REPORT and sampler control statements, which are detailed in the <i>CMF Monitor Batch User Guide and Reference</i>. This member controls the operation of the IPM mode samplers and defines the resources the Extractor monitors.</p> <p>IPM mode samplers execute either when a MODIFY command is issued or when the IPM parameter is used. Sample control statement members are in the <i>hilevel</i>.UBBPARM data set created by AutoCustomization.</p> <p>The IPM mode control statement member the Extractor uses can be changed while the Extractor is active by using the MODIFY command; see <i>MAINVIEW Administration Guide</i> for more information.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
XDS={xx}STOP}	<p>Optional (CMF MONITOR only). Controls the collection of data by the cross-system data server (XDS) (MVS 5.1 and above and MVS 4.3 with Dynamic Exit Facility usage restrictions).</p> <p>To share cross-system data, all the systems must be in the same sysplex, the MVS PAS on each system must have XDS active, DC=START must be specified for each MVS PAS, and the CAS on all systems must be in the same CASDEF XCF group.</p> <p>For SDSF to use XDS data, the BBLINK data set needs to be in either the link list or in the TSO logon procedure STEPLIB.</p> <p>To collect XDS data, specify XDS=xx, where xx indicates the suffix of a CMFXDSxx member of <i>hilevel</i>.UBBPARM created by AutoCustomization. You can switch to a different XDS member without stopping either the MVS PAS or XDS by using the MODIFY command. For example, if you are running the MVS PAS pointing to the CMFXDS00 member with XDS=00, you can begin using the CMFXDS01 member by issuing the MODIFY command <code>F MVSPAS ,XDS=01</code>. You can also stop XDS without stopping the MVS PAS by issuing the MODIFY command <code>F MVSPAS ,XDS=STOP</code>.</p> <p>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.</p> <ul style="list-style-type: none"> • TYPE <p>Defines which SMF record types are included in the XDS SMF record buffer for the CX10XDQY and CX10XDRC APIs. Specify one of the following:</p> <p>TYPE CMF to use output records from CMF itself. Use this to buffer all type 70-78 records, as well as all CMF user record types. This is the default value if the TYPE parameter is not specified.</p> <p>TYPE SMF to use SMF IEFU83, IEFU84, and IEFU85 dynamic exits. Use this 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.</p> <p>TYPE ALL to use both SMF-provided and CMF-provided records.</p> <p>TYPE NONE to allow API calls and CX10XDGS requests for type 79 data from this system without buffering records on this system.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
<p>XDS={xx STOP} (continued)</p>	<ul style="list-style-type: none"> • RECORDS <p>Defines which SMF record types and subtypes are included in the XDS buffer. This must be a subset of the records specified for collection by the TYPE parameter. Records specified with this parameter are available for CX10XDQY (XDS record query) and CX10XDRC (XDS record retrieval) API calls.</p> <p>In addition, cross-system snapshots of type 79 data are available from the CX10XDGS (XDS data-gathering service) API call.</p> <p>Specify the records 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. Subtypes can be specified 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.</p> <p>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.</p> <p>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 is the default if you do not include a RECORDS parameter.</p> <ul style="list-style-type: none"> • SIZE <p>Defines how much SMF data should be buffered before records are lost because of buffer wrap-around.</p> <p>Specify a size in the format of SIZE <i>nnnu</i>, where <i>nnn</i> is a one- to nine-digit number and <i>u</i> is one of the following:</p> <p>(blank) bytes K kilobytes (2**10 bytes) M megabytes (2**20 bytes) G gigabytes (2**30 bytes)</p> <p>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.</p> <p>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.</p>
<p>EM={xx 00}</p>	<p>(MAINVIEW for OS/390 only.) Specifies the two-character suffix of the PWSCPMxx control statement member you want MAINVIEW for OS/390's Exception Monitor to use.</p>

Table B-2 MVS PAS PROC Statement Parameters (continued)

PROC Parameter	Description
SSID= <i>ssid</i>	<p>Required; specifies the subsystem ID of the CAS to which the MVS PAS should connect. The CAS is started as a separate address space from the MVS PAS and must be initialized before the MVS PAS is initialized.</p> <p>For a PAS to connect to a CAS, the SSID values in both the CAS and PAS startup procedures must be the same.</p>
CMDID= <i>symbol</i>	<p>Optional; defines a single character that can be used in place of the MODIFY command and <i>procname</i> when a MODIFY command is issued to the PAS. You can use the CMDID= parameter as a quick method for issuing commands.</p> <p>You must manually add the CMDID= parameter to the PAS procedure statement to enable this support.</p> <p>Valid operands are:</p> <pre> ¢ . (+ & ! `) &#224 ! " - / % _ > ? @ * = # < </pre> <p>Certain operands may need to be enclosed within quotes to work in your specific environment. Alternatively, operands may be specified as a 2-digit hexadecimal representation.</p> <p>For example, if you specify CMDID=* on the PAS procedure statement, you can issue the command DC=STATUS from the console by entering This uses fewer keystrokes than issuing the MODIFY command, but either method can be used. For information about using the MODIFY command, see the <i>MAINVIEW Administration Guide</i>.</p>
BBCHILV=	Required; defines the high-level qualifier for the product libraries.
BAVSHLQ=	Required; defines the high-level qualifier for the VSAM data sets used for historical data and internal and user-defined system workloads
UBBPARM=	Required; defines the name assigned to the user parameter library.
BBLINK=	Optional; defines the name assigned to BBLINK. If BBLINK is in your system link list, this parameter and the //STEPLIB DD statement should be removed.
BBSOUT=	Defines a valid SYSOUT class for started tasks.
RGN=32	Assigns a region size of 32 megabytes.

Appendix C Library Members for BBI-SS PAS

This section describes the parameters provided by some BBPARM members. These parameters affect the BBI-SS PAS product environment. Use BBSAMP member SSJCL to help you set up your BBI-SS PAS started task procedure.

BBPARM Member BBIBBCFG Parameters

BBPARM member BBIBBCFG is a model for UBBPARM members to be used by all Group 1 and Group 3 products to identify suffixes of BBPARM configuration members to the BBI-SS PAS.

By using the BBCFG facility, multiple BBI-SS PASs can process different BBPARM configuration members even when they are sharing the same BBPARM data set.

The following table lists the parameters for BBIBBCFG.

Table C-1 BBPARM Member BBIBBCFG

Parameter	Description
AAOALS=00	(MAINVIEW AutoOPERATOR only) AAOALSxx members contain information that control how much storage MAINVIEW AutoOPERATOR ALERTs use. The suffix can be any two characters that are a valid partitioned data set member name.
AAOARP=00	(MAINVIEW AutoOPERATOR only) AAOARPxx members contain parameters that control how the MAINVIEW AutoOPERATOR Automation Reporter application operates. The suffix can be any two characters that are a valid partitioned data set member name.
AAOEXP=00	(MAINVIEW AutoOPERATOR only) AAOEXPxx members contain parameters and thresholds that determine how efficiently MAINVIEW AutoOPERATOR EXECs execute on your system. The suffix can be any two characters that are a valid partitioned data set member name.
AAOGME=00	(MAINVIEW AutoOPERATOR only) AAOGMExx members contain parameters that MAINVIEW AutoOPERATOR uses to create General Messages Exchange (GME) connections to allow applications to communicate.
AAOMQL=00	(MAINVIEW AutoOPERATOR only) AAOMQLxx members contain parameters that MAINVIEW AutoOPERATOR for MQSeries uses to specify which MQ queues are listened to by MAINVIEW AutoOPERATOR for automation.
AAOPRM=00	(MAINVIEW AutoOPERATOR only) Is the BBPARM member suffix used by the BBI-SS PAS. The suffix can be any two characters that are a valid partitioned data set member name.
AAOTSP=00	(MAINVIEW AutoOPERATOR only) AAOTSPxx members contain parameters that control how the MAINVIEW AutoOPERATOR TapeSHARE application operates. The suffix can be any two characters that are a valid partitioned data set member name.
BBISSP=00	BBISSPxx members contain the BBI-SS PAS subsystem control parameters. The suffix can be any two characters that are a valid partitioned data set member name.

BBPARM Member BBIISP00 Parameters

BBPARM member BBIISP00 contains parameters that describe the initialization of timer-driven services. For other product-specific parameters, see the comment field of the BBIISP00 member.

Table C-2 BBPARM Member BBIISP00

Parameter	Description
MAXREQ= <i>n</i>	(Optional) Where <i>n</i> is the maximum number of concurrent timer-driven requests. The default is 500. The maximum value is 3000.
INTERVAL= <i>hh:mm:ss</i>	(Optional) Default timer interval when one is not supplied by the request itself. The default is 00:01:00 (one minute).
IRRI= <i>hh:mm</i>	(Optional) Specifies a recording interval in hours and minutes when records are to be written to a historical data set. The default is 15 minutes.
AUTOID= <i>xxxxxxxx</i>	(Optional) Specifies the logon ID of a user to be associated with automatically started timer requests. It is the default if USERID is not specified. AUTOID is written on a separate line from the TARGET statement.
TARGET= <i>xxxx</i> , BLK= <i>xxxxxxxx</i> , [USRID= <i>xxxxxxxx</i>]	TARGET specifies the target system to receive the requests for BLK. It can be either the target system ID or the subsystem ID. TARGET is written on the same line as BLK and can be repeated. BLK specifies the name of a member containing multiple timer-driven requests to be invoked at BBI-SS PAS startup. This member must exist in the BBPARM data set. BLK is written on the same line as TARGET. (Optional) USRID specifies the logon ID of a user to be associated with the timer requests in the member specified with the BLK parameter. It is written on the same line as TARGET and BLK. If AUTOID is specified, USRID supersedes AUTOID.
TRDIR= <i>xxxxxxxx</i> SUBSYS= <i>ssid</i>	(Optional - MAINVIEW for CICS, MAINVIEW for IMS Online, MAINVIEW for DBCTL, and MAINVIEW for DB2) TRDIR specifies the name of the data set used for a trace directory (required for trace logging). (Optional) SUBSYS specifies the subsystem name to which this trace directory applies. This parameter allows multiple BBI-SS PASs to share the same BBIISP00 member.

BBPARM Member BBISSPxx Parameters

BBPARM member BBISSPxx contains parameters that control a BBI-SS PAS.

BBISSPxx Parameters for the BBI-SS PAS Product Environment

The following table describes the BBISSPxx parameters used by products that run in the BBI-SS PAS.

Table C-3 BBPARM Member BBISSPxx - BBI-SS PAS Control Parameters

Parameter	Description
CASID=xxxx	Specifies a 1- to 4-character ID of the MAINVIEW coordinating address space (CAS).
CMDCHAR=x	(Optional) Defines the command characters to be used for issuing control commands from an MVS console to this subsystem. If CMDCHAR is not specified, control commands can still be issued from a console with the MVS modify command. This parameter is optional but if CMDCHAR is specified, it must not be a blank; CMDCHAR= is not valid.
DAE=[YES NO]	Specifies whether dumps with duplicate symptom strings are suppressed when DUMPS=SDUMP is specified. Only specify DAE=NO in controlled diagnostic situations when instructed to do so by BMC Software Customer Support.
DUMPCLAS=x	Output class for formatted dumps. The default value for x is A.
DUMPS=[YES NO SDUMP ALL]	Specifies whether formatted dumps are to be taken in case of a failure: If YES is specified, a BBI-formatted dump is created. If NO is specified, only LOGREC records are written and no SVC dump or BBI-formatted dump is created. If SDUMP is specified (default), SVC dumps are created and DAE=YES is used to suppress duplicate SVC dumps. For MAINVIEW AutoOPERATOR-related dumps, you should use DUMPS=SDUMP. If ALL is specified, both an SVC dump and a BBI-formatted dump are created.
JESCNCT=[YES NO]	(Optional) Specifies whether the BBI-SS PAS should attempt to connect to the MVS job entry system (JES). This parameter is only valid when the PAS is started <ul style="list-style-type: none"> during system IPL before JES is started outside the control of JES using SUB=MSTR on the MVS START command <p>If YES is specified, the BBI-SS PAS attempts to connect to JES as soon as it is available. If the connection is successful, the BBI-SS PAS is given</p> <ul style="list-style-type: none"> a JES job ID a JES job log the ability to use JES dependent services such as SUBMIT processing or SYSOUT=x for the allocation of subsystem-specific data sets (if MAINVIEW AutoOPERATOR is present) <p>If NO is specified (default), no attempt is made to connect to JES and no JES services are available to the BBI-SS PAS.</p>

Table C-3 BBPARM Member BBISSPxx - BBI-SS PAS Control Parameters (continued)

Parameter	Description
JOURNAL=[STD STANDARD ENHANCED]	(Optional) Determines if subsets of the journal can be displayed: If STD or STANDARD is specified (default), the entire journal is always shown in the Log Display application. If ENHANCED is specified, you can use Log Display PROFILE specifications to display subsets of the journal.
JRNLCMSG= <u>NO</u> YES	Specifies whether split BBI-SS PAS Journal log messages are to be identified in the BBIPRINT journal log print data set. YES is the default. Split messages are identified by a 1-character prefix as follows: 1 First line of a split message + Continuation line of a split message - Ending line of a split message
JRNLCSTC= <u>BBIDLOG</u>	Specifies a 1- to 8-character name of a started task used to dump and reset the BBI-SS PAS Journal logs. The default is BBIDLOG.
LOGGING=[<u>YES</u> NO]	Specifies whether BBI-SS PAS journaling is to be performed. The default is YES.
LOGON=[YES <u>NO</u> USS]	Determines if and how terminal session LOGONs are accepted by this BBI-SS PAS. USS means that LOGONs are accepted by the USS table instead of by the LOGON interpret table, which is the default when YES is specified.
MAXPQE= <i>nnnn</i>	Maximum number of PQEs used concurrently in CSA between the target and the BBI-SS PAS. The default is 300. The maximum number of concurrent PQEs is 9999.
MAXMQE=35	MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS version 5.3.01 or earlier only. Maximum number of MQEs used by MAINVIEW AutoOPERATOR for CICS when invoking MAINVIEW for CICS services. MQEs represent concurrent actions that can be taken by MAINVIEW AutoOPERATOR for CICS; for example, issuing multiple OPEN/ENABLE commands. Do not use this parameter unless the current amount of output from these commands is not sufficient. Each additional MQE over and above the default of 35 uses 8 bytes of CSA below the line. The maximum value allowed is 999.
MAXUSER= <i>nn</i>	Maximum number of concurrent terminal sessions that can access this BBI-SS PAS. The default is 20.
PRODUCT= <i>xxx</i>	The value <i>xxx</i> represents the code of a product to be initialized within this BBI-SS PAS. Valid values are AAO, CMR, DBC, DMR, IMF, MFP, MVMQS, and MVVP.
STALL= <i>nnn</i>	Maximum number of seconds to complete the initialization process. If this is exceeded, the BBI-SS PAS fails to initialize and issues a STALL DETECTED message. The is 180. A value of 0 sets the stall limit to 8 hours.

BBISSPxx Parameters for MAINVIEW AutoOPERATOR Only

The following table describes the BBISSPxx parameters used only for MAINVIEW AutoOPERATOR.

Table C-4 BBPARM Member BBISSPxx - MAINVIEW AutoOPERATOR Parameters

Parameter	Description
ALRTRCVE=[YES NO]	<p>Specifies whether this BBI-SS PAS is to be considered a product alert receiver.</p> <p>If YES is specified, the PAS can receive product alerts and alarm messages from other BMC Software products.</p> <p>If NO is specified (default), the PAS cannot receive alerts from other products. The PAS can, however, process alerts created by MAINVIEW AutoOPERATOR EXECs, rules, or AOAnywhere calls in this BBI-SS PAS.</p> <p>To determine if a PAS is an alert receiver, issue the following BBI command:</p> <pre>.D XCF</pre> <p>If message AB3006I is displayed, the PAS is an alert receiver.</p>
ARMPOLCY[YES NO xxxxxxx]	<p>Indicates whether the MVS/ESA Automatic Restart Manager (ARM) function is specified. Valid values are:</p> <p>NO Specifies that the BBI-SS PAS will not use ARM.</p> <p>YES Specifies that the BBI-SS PAS will use the ARM default restart policy. The default ARM restart policy is that ARM restarts an abended address space up to 3 times within 5 minutes. If the address space fails to restart after the third attempt, a message is written to the SYSLOG and the address space remains unstarted.</p> <p>xxxxxxx Specifies a 1- to 8-character name of a user-created policy to be used to control BBI-SS PAS restarts.</p> <p>For more information about implementing ARM support for MAINVIEW AutoOPERATOR, see the <i>MAINVIEW AutoOPERATOR Customization Guide</i>.</p>
AUDIT=[YES NO]	<p>Indicates whether rule set updates and time-initiated setups and purges are recorded in the Journal. The default is NO.</p> <p>The following are not monitored with AUDIT=YES:</p> <ul style="list-style-type: none"> • Time-initiated EXEC SETs and PRGs issued from the TI application • SETs and PRGs issued from the TI application
CMDCON= <i>id</i>	<p>A 2-byte console ID (from 1 to 99). Must be an active, valid console that is available to MAINVIEW AutoOPERATOR and cannot be used by another address space. The default is no console ID.</p> <p>Alternatively, a 2- to 8-character multiple console support (MCS) console name can be specified. MCS console names are defined in SYS1.PARMLIB member CONSOLxx and are generally more meaningful in a sysplex environment. Extended MCS (EMCS) console IDs and names are not supported.</p>
CMDWAIT=xx	<p>Defines the number of seconds to wait for a response to an MVS command issued by a terminal session user. The default is 3.</p>

Table C-4 BBPARM Member BBISSPxx - MAINVIEW AutoOPERATOR Parameters (continued)

Parameter	Description
CNSLEXEC=[YES NO]	Indicates whether operators can invoke EXECs through the console interface (MVS modify command or subsystem character command). Default is YES.
HILITE[YES NO]	Indicates whether IMS messages, beginning with a /, should be highlighted in the BBI-SS PAS Journal.
JESFLTR=CMD WTO	Specifies what function is to filter subsystem requests before JES. If CMD is specified, Rules can suppress JES commands. If WTO is specified, messages to a JES3 console can be suppressed. You can combine the two values like this: JESFLTR=(CMD,WTO). You must specify a valid CMDCHAR parameter for the UNHOOK command to work.
MCTSIZE=510	Specifies how many unique message identifiers to track for the Message Statistics application. Valid range is 0 - 9999. Specifying a value of zero completely disables the gathering of message statistics.
XCFGROUP= <i>name</i>	Specifies the 1- to 8-character name of a cross-system coupling facility (XCF) group that this BBI-SS PAS is to join. The default is BMCAB. The XCF group that you specify should be used only for BBI-SS PASs; otherwise, communication with other subsystems will be adversely affected. Only BBI-SS PASs that belong to the same XCF group and reside in the same sysplex can communicate with each other. To determine what XCF group a PAS belongs to, issue the following BBI command: .D XCF
MAINVIEW AutoOPERATOR for Access NV Option Only	
NETVIEW= <i>name</i>	Specifies the 1- to 8-character name of the NetView subsystem to which the MAINVIEW AutoOPERATOR subsystem is to connect.
MAINVIEW AutoOPERATOR for OS/390 Option Only	
CONSOLES= <i>nn</i>	Represents the number of X-MCS consoles (without MIGID) to be acquired by MAINVIEW AutoOPERATOR. The default is 4; possible values are 0 through 99. The number represents the number of concurrent commands that can execute.
MIGCONS= <i>nn</i>	Represents the number of additional X-MCS consoles to be acquired by MAINVIEW AutoOPERATOR. The default is 1; possible values are 0 through 99. Migration IDs are 1-byte console IDs. MVS Version 4 non-sysplex users can force MAINVIEW AutoOPERATOR to use X-MCS consoles by using this parameter.
JESCHAR=\$	JES2 control command character.
JES3CHAR=	Character to recognize as JES3 command character. This is ignored when JES3CMD=N is specified, since the SVC34 interface is used. The default is *.
JES3CMD=[Y N]	Specifies whether the subsystem interface should be used to issue JES3 commands. This should be set to N for JES 3 Release 2.2.1 and above. The default is Y.

BBISSPxx Parameter for MAINVIEW for CICS, MAINVIEW for IMS Online, MAINVIEW for DBCTL, and MAINVIEW for DB2

The IMAGSTC parameter in BBISSPxx applies only to these products.

Table C-5 BBPARM Member BBISSPxx - CICS, IMS Online, DBCTL, and DB2 Parameter

Parameter	Description
IMAGSTC= <u>BBILOG</u>	Specifies a 1- to 8-character name of a started task used to print the image logs. The default is BBILOG.

BBISSPxx Parameter for MAINVIEW AutoOPERATOR for MQSeries and MAINVIEW for WebSphere MQ Integrator

The GTS parameter in BBISSPxx applies only to these products.

Table C-6 BBPARM Member BBISSPxx - MQSeries Parameter

Parameter	Description
GTS=[NO <i>nn</i> 00]	Controls whether the Generic TCP/IP Server (GTS) is started during BBI-SS PAS initialization. NO The GTS is not started. <i>nn</i> Indicates the 2-character suffix of the BBPARM member BBTTC <i>nn</i> and starts the GTS using that member. If the GTS parameter is omitted, BBTTC00 is used to start the GTS.

BBPARM Member BBTTC00 Parameters

BBPARM member BBTTC00 contains parameters that control whether a BBI-SS PAS supports the BMC Software TCP/IP interface used by MAINVIEW AutoOPERATOR, MAINVIEW AutoOPERATOR for MQSeries, and the MAINVIEW Explorer host server. The following table describes the BBTTC00 parameters:

Table C-7 BBPARM Member BBTTC00

Parameter	Description
TCPNAME= <i>name</i>	<p>Specifies the name of the TCP/IP address space. For TCP/IP Release 3.2 or above, you can omit the TCPNAME parameter and the BBI-SS PAS will attempt to determine the TCP/IP address space to use. If more than one address space is found, the first one found with the latest release is used.</p> <p>Make sure the TCP/IP address space specified with the TCPNAME parameter is of the same release as the TCP/IP stack specified with the STACK parameter (described below). Failure to do so can result in degraded performance or error messages.</p>
STACK=[NONE IBM31 IBM32 IBM34 ILINK41 ILINK52]	<p>Specifies the TCP/IP stack as one of the following:</p> <p>NONE Disables TCP/IP processing.</p> <p>IBM32 For IBM TCP/IP release 3.2</p> <p>IBM34 For IBM TCP/IP release 3.4</p> <p>ILINK41 For Computer Associates SOLVE:TCPaccess release 4.1</p> <p>ILINK52 For Computer Associates SOLVE:TCPaccess release 5.2</p> <p>For optimal performance, specify a TCP/IP address space and stack that are of the same release.</p> <p>For TCP/IP, you can omit the STACK parameter and the BBI-SS PAS will attempt to determine the TCP/IP stack to use. If more than one stack is found, the first one found with the latest release is used.</p>
INFOMSG=[YES NO]	<p>Determines whether TCP/IP interface startup messages are displayed on the console. The default is YES.</p>
DEBUG=[ALL ERROR PARM xxxxxx]	<p>Warning: Specify this parameter only at the request of BMC Software Customer Support, as it causes many messages to be written to the operator console.</p> <p>Activates debug mode and writes diagnostic messages to the operator console. You can specify multiple DEBUG statements, provided they do not conflict with one another.</p> <p>ALL Produces a message for each socket call.</p> <p>ERROR Produces messages only for unsuccessful socket calls.</p> <p>PARM Produces a message with parameters 4 through 9 for each socket call.</p> <p>xxxxxx Produces only messages for the specified task ID, where xxxxxx is a task ID. Each MAINVIEW product has one or more task IDs. The main TCP/IP task ID is "Global". Refer to your MAINVIEW product documentation for product-specific task IDs.</p> <p>Debug messages are documented in the MAINVIEW online message system. All debug messages have a suffix of D. Some MAINVIEW products may choose to display these messages even when debug mode is not active. Those messages will have a suffix of I or E. For those products, if debug mode is activated, some messages will be issued twice, once with the I or E suffix, and once with the D suffix.</p>

BBPROF Data Set Members

The BBPROF data set contains profile information and cycle refresh definitions for a terminal session. Other BBPROF members are dynamically created. These members should not be edited nor deleted because they contain important profile information per BBI-SS PAS application.

Table C-8 BBPROF Data Set

BBPROF Member Name	Description
xxxCYC CYCnn	<p>Where xxx is an application prefix. Members in this data set with names beginning with xxxCYC contain sample definitions of cycles for refreshing services.</p> <p>Other members can be added to BBPROF using a naming convention of CYCnn or using any valid member name. In the second case, the definition is activated with the full member name.</p> <p>A CYCnn definition can be activated in the CYCLE application with a SET nn command, where nn specifies an alphanumeric suffix.</p>
BBICMDS	<p>This member contains the BBI control command table (see the <i>MAINVIEW Administration Guide</i> for a description of the BBI control commands).</p>
BBIPFK	<p>This member contains default program function key definitions that are used if a xxxxPFK member is not defined for a product application; for example, IMSPFK.</p>
BBITSP00	<p>This member contains profile information for a terminal session. It defines the primary product application (PRIMAPPL) to display when the terminal session starts if multiple application types, target names, and associated menus are specified:</p> <p>TARGET=tgtname, TYPE=appltype[, PRIMMENU=name]</p> <p>The application type can be AO, IMS, CICS, or DB2 depending upon the product installed. Other parameters define:</p> <ul style="list-style-type: none"> • Whether local logging (terminal session journaling) is performed (LOGGING=YES). This parameter is set to NO if BBIJRN1 DD is not allocated in the TSCLIST (for example; JOURNAL(YES) in TSCLIST). • The default screen refresh interval (INTERVAL) • Screen characteristics (SCREEN) • Characters (CAPS) • Colors (COLORS) • Reverse video (REVERSE) • Formatted dump output (DUMPS and DUMPCLAS) • ISPF split-screen (ISPF)

Table C-8 BBPROF Data Set (continued)

BBPROF Member Name	Description
LOGPFK	This member contains default program function key definitions for the General Services LOG DISPLAY option.
xxxxPFK	<p>These members contain program function key definitions. The member name prefix can be IMS, DB2, MVS, or CICS.</p> <p>The definitions are displayed when the KEYS option is selected from the Primary Option Menu or when the KEYS command is entered. For more information, see the <i>Using MAINVIEW</i> manual.</p> <p>The definitions can be changed online.</p> <p>If the member contains the MASTER keyword, the online changes cannot be stored in the xxxxPFK member when the SAVE command is entered from the COMMAND line.</p> <p>If the MASTER keyword is removed and there is no user BBPROF data set, subsequent online PF key changes are saved in the PF key member of the site's BBPROF data set.</p>

Appendix D Library Member for MAINVIEW Alarm Manager PAS

Table 4-1 describes the MAINVIEW Alarm Manager PAS startup JCL statements. These statements can be found in BBSAMP member BBHPAS.

Table 4-1 MAINVIEW Alarm Manager PAS JCL Statements

JCL Statement	Description
// <i>procname</i> PROC	See Table 4-2 on page D-2 for an explanation of each PROC statement parameter. You can override these parameters with the START command for the PAS, as described in the <i>MAINVIEW Administration Guide</i> .
//PAS EXEC	Specifies the program name (PGM=SSLOAD). The PARM= subparameters are all symbolic and use the same values defined for the corresponding parameter in the PROC statement. For a detailed explanation of each parameter, see Table 4-2 on page D-2.
//STEPLIB DD	Defines a partitioned data set that contains the MAINVIEW Alarm Manager load modules. The partitioned data set must be an authorized library, such as <i>hilevel.BBLINK</i> . This statement should be deleted if BBLINK was added to your system link list (see the <i>MAINVIEW Administration Guide</i>).
//BBACTDEF DD	Defines the library containing all action and view tables for MAINVIEW Alarm Manager.
//BBIPARM DD	Contains the BBHTMNxx members, where xx is the ID for the group. Each member contains information for all alarm definitions in the group.

Table 4-1 MAINVIEW Alarm Manager PAS JCL Statements (continued)

JCL Statement	Description
//BBSECURE DD	<p>Defines the data set that contains MAINVIEW Alarm Manager security resource definitions. These definitions are used by the CAS and PAS to determine if security checking should occur for system, product, view, and/or action access. The same data set name must be defined for the CAS started task procedure BBSECURE statement and the PAS started task procedure BBSECURE statement.</p> <p>Defining a BBSECURE data set is only one part of setting up security to function with your external security manager (ESM). For more information, see <i>Implementing Security for MAINVIEW Products</i>.</p>
//BBVDEF DD	Defines the view library containing all of the default views for MAINVIEW Alarm Manager.
//BBIJRN1 DD //BBIJRN2 DD	<p>Defines the journal log data sets that MAINVIEW Alarm Manager uses to record</p> <ul style="list-style-type: none"> • commands and responses issued by the Alert Management component • product alerts, warnings, and messages
//BBIDIV DD	Defines the data-in-virtual data set that is used to store binary large objects and data across MAINVIEW Alarm Manager restarts.

Table 4-2 describes the MAINVIEW Alarm Manager PAS PROC statement parameters.

Table 4-2 MAINVIEW Alarm Manager PAS PROC Statement Parameters

PROC Parameter	Description
MVALSS	Specifies a 1- to 4-character subsystem ID for the MAINVIEW Alarm Manager started task procedure. This ID must be unique within the MVS image where MAINVIEW Alarm Manager will be started.
START= <u>WARM</u> COLD FREE	<p>Specifies the type of MAINVIEW Alarm Manager PAS startup. WARM is the default and should be used whenever possible.</p> <p>A COLD start may occur automatically after the system is IPLed or maintenance is applied. Otherwise, do not request a COLD start or FREE start unless you are instructed to by BMC Software Customer Support.</p>
MVAJRN1	<p>Identifies the MAINVIEW Alarm Manager journal log data sets.</p> <p>The data set names are derived by adding a suffix of 1 or 2 to the value of this parameter. For example, if you specify MVAJRN1='BMC.MVAJRN1', the started task procedure looks for two data sets called 'BMC.MVAJRN11' and 'BMC.MVAJRN12'.</p> <p>The data sets you identify here must be allocated using the //BBIJRN1 and //BBIJRN2 DD statements in the PAS startup JCL.</p>
MVADIV	<p>Identifies a data set to be used by MAINVIEW Alarm Manager for storing data across product restarts.</p> <p>The data set you identify here must be allocated using the //BBIDIV DD statement in the PAS startup JCL.</p>

Table 4-2 MAINVIEW Alarm Manager PAS PROC Statement Parameters (continued)

BBLINK	Optional; defines the name assigned to BBLINK. If BBLINK is in your system link list, this parameter and the //STEPLIB DD statement should be removed.
BBCHILV	Required; defines the high-level qualifier for the product libraries.
MVAPARM	Required; defines the name assigned to the MAINVIEW Alarm Manager user parameter library.

Appendix E Implementing EXCP or VTAM Terminal Sessions

All MAINVIEW products include a component called MAINVIEW Alternate Access that provides EXCP and VTAM communication to these products through ISPF without requiring a TSO subsystem to be active. MAINVIEW Alternate Access fully supports:

- EXCP or VTAM terminal sessions (including the MAINVIEW AutoOPERATOR application CSM and RxD2)
- All of the existing and new MAINVIEW product releases

Note: MAINVIEW Alternate Access requires ISPF. If ISPF is not available in your environment, MAINVIEW Alternate Access cannot be used.

Using AutoCustomization

You should use AutoCustomization to customize MAINVIEW Alternate Access. The required steps appear at the end of the shared step selection list provided by AutoCustomization and show BBV as the product code.

If you decide not to use AutoCustomization, see the *MAINVIEW Alternate Access Implementation and User Guide* for manual customization instructions.

Note: Changes made to the MAINVIEW CLIST support the MAINVIEW Alternate Access feature AUTOLOGON. Following AutoCustomization of MAINVIEW CLIST, review and replace any copies of TSCLIST you may have already made.

Deactivating Previous VTAM Support

Once you have completed the implementation of MAINVIEW Alternate Access, you should deactivate any previous VTAM support provided by the products shown in Group 1 of Table 4-1 on page 4-2.

To do this, change the LOGON parameter setting in BBPARM member BBISSP00 from:

LOGON=YES

to:

LOGON=NO

Index

Symbols

@@YZ021 5-23
@@YZ061 5-38, 5-44
@@YZ062 5-44
@@YZZ001 5-7
@@YZZ011 5-12
@@YZZ021 A-1
@@YZZ051 5-36, 5-41
@@YZZ052 5-36, 5-41
@@YZZ061 5-38
@@YZZ081 5-22

A

ACF2 5-112
action table A-2, B-4
ADD command for CAS 5-96
adding

- BBI-SS PAS 5-80
- CAS definitions 5-95
- load library to link list 5-16

Alarm Manager, see MAINVIEW Alarm Manager
ALIAS 5-50
allocating site data sets 5-10
ALRTRCVE parameter C-6
Alternate Access, MAINVIEW 1-5
APF-authorizing load library 5-14
APPLID 5-83, 5-84
ARM, enabling/disabling 5-103, C-6
ARMPOLCY parameter C-6

ASMLKED 5-66, 5-67
AUDIT parameter C-6
authorized command processor table 4-4, 5-111
authorizing

- BMC Software load library 5-14
- products to run under ISPF 5-20

AUTHTSF 5-20
AutoCustomization 1-6

- library 3-5
- parameter library 3-7
- procedure library 3-12
- profile library 3-10
- sample library 3-9

AUTOID parameter C-3
Automatic Restart Manage, enabling/disabling C-6
Automatic Restart Manager, enabling/disabling 5-103
AutoOPERATOR

- event management utility 4-5
- EXECs 5-62
- rule sets 4-5

AutoRule 4-5

B

BAIAPF 5-14
BAIMAINV 5-18
BALCMSG 5-111
BAVSHLQ B-10
BBACTDEF 5-13, A-2, B-4, D-1
BBAIVP 5-4
BBATSP00 5-113

BBCFG 5-51
BBCFG facility C-2
BBCHILV B-10
BBCLIB library 3-5
BBCSJ008W 5-98
BBCSJ015W 5-98
BBGROUP 5-94
BBHPAS 5-33
BBI
 architecture 1-2
 Command Table 5-71
 release 2-1
BBIBBCFG 3-16, 5-51, C-2
BBICMDS 5-71, C-10
BBIDIV 3-17
 MAINVIEW Alarm Manager 5-32
 MAINVIEW AutoOPERATOR 5-62, 5-64
BBILOG C-8
BBIISP00 5-53, C-3
BBIJNT00
 dynamic targets 5-46
 use 5-48, 5-83
BBILIB member @@YZZ021
 BBACTDEF statement A-2
 BBIPARM statement A-2
 BBSECURE statement A-2
 BBVDEF statement A-2
 EXEC statement A-1
 PROC statement A-1
BBILIB member BBMPAS
 BBACTDEF statement B-4
 BBVDEF statement B-4
 CMFCDS statement B-3
 CMFCPM statement B-2
 CMFDUMP statement B-3
 CMFIPM statement B-2
 CMFLOG statement B-3
 CMFMSG statement B-3
 CMFXDSLJ statement B-3
 EXEC statement B-1
 PARMLIB statement B-3
 PROC statement B-1
BBINOD00, use 5-83
BBIPARM A-2
BBIPFK C-10
BBIRESP 5-71
BBI-SS PAS 5-72
 adding 5-80
 ALIAS 5-50
 ARM 5-103
 CAS ID 5-109
 communication 5-72, 5-81
 DESC 5-50
 dispatching priority 5-59
 extended functions 5-110
 historical data set 5-41
 IMSID 5-50
 IMSTYPE 5-49
 Journal logs 5-57
 log handling procedure 5-65
 PARMFILE 5-44
 PREFIX 5-60
 product initialization 5-102
 region size 5-59
 RELEASE 5-50
 started task procedure 5-59
 SUBSYS 5-49
 subsystem ID 5-59
 suffixes 5-46
 TARGET 5-49
 timer-driven services 5-53
 TYPE 5-49
BBISSP00
 ARM 5-103
 CAS ID 5-109
 MAINVIEW Alarm Manager 5-28
 PRODUCT parameter 5-28, 5-29, 5-102
 use 5-65, 5-102
BBISSPxx
 parameters C-4 to C-8
BBITSP00
 description of C-10
 use of 5-69
BBIUSR01 5-65
BBIUSR02 5-67
BBIVARS 3-17, 5-57
BBIXSP00 5-83
BBKEYS 5-104, 5-106
BBLINK B-10
 allocating to the CAS A-1
 allocating to the PAS B-1
 APF authorization 5-14
 TSF authorization 5-20
BBM3API 5-111
BBM9DA00 B-1
BBM9TC21 5-111

BBM9TC22 5-111
 BBM9TC24 5-111
 BBM9ZA00 A-1
 BBMCAS 5-23, 5-24, A-2
 BBMLIB
 description 3-4
 BBMPAS 5-39, B-1
 BBMTYB00 member 5-27
 BBMXPJCL 5-25
 BBPARM
 BBIBBCFG member C-2
 BBIISP00 member C-3
 BBISSPxx member C-4
 BBTTCP00 member C-9
 considerations 3-1
 BBPLIB
 description 3-5
 BBPROF
 considerations 3-1
 data set members C-10
 use 5-69
 BBSAMP
 considerations 3-1
 BBSDEF 3-13, 3-14
 BBSECURE
 creating 5-13
 customizing A-2, B-4, D-2
 BBTLIB
 description 3-5
 BBTTCP00 5-26
 BBTTCP00 member C-9
 BBVDEF 3-13, A-2, B-4, D-2
 BBVJSETP 5-111
 BBXS 5-21
 BBXSINIT 5-21
 BLK parameter C-3
 BLKDBTW 5-54
 BLKDMRW 5-54
 BLKIMFW 5-54

C

CA-ACF2 5-111
 CAPS parameter C-10
 CAS, see coordinating address space
 CASDEF view 5-95 to 5-98
 CASID parameter C-4
 CDRM definitions 5-78
 CICS@MR 5-71
 CICS@MVA 5-71
 CLIST
 library
 description 3-5
 CMDCHAR parameter
 BBISSPxx member C-4
 CMDCOM parameter C-6
 CMDWAIT parameter C-6
 CMRBLK1 5-54
 CMRBLK2 5-54
 CMRBLK3 5-54
 CNSLEXEC parameter C-7
 COLD start
 BBI-SS PAS 5-60
 CAS A-3
 COLORS parameter C-10
 command authorization table 4-4, 5-111
 Command Table
 BBI
 BBICMDS 5-71
 ISPF TSO 5-111
 communication 5-72
 between address spaces 1-3
 communication links
 CASs 5-75, 5-77
 XCF 5-75
 configuration
 BBI-SS PAS 5-51
 data sets 3-1
 MAINVIEW AutoOPERATOR 3-17
 product libraries 3-2
 CONSOLES parameter C-7
 CONVXCF CAS PROC parameter A-3
 coordinating address space
 adding ID to BBI-SS PAS 5-109
 adding system definition 5-95
 communication 5-72
 cross-system considerations 5-75
 defining ID A-2
 description 1-3
 migrating definitions 5-27
 START command 5-24
 started task procedure statements A-1
 startup procedure 5-23
 VTAM definitions 5-75
 CPM

- mode samplers B-7
- monitor B-2, B-7
- CPU, changing or adding 5-108
- cross-domain
 - BBI-SS PAS, defining 5-84
 - CAS, defining 5-91
 - communication 5-81
- cross-system communication 5-72
 - BBI-SS PAS
 - defining 5-82
 - planning 5-79
 - CAS
 - considerations 5-75
 - defining 5-86
 - planning 5-75
 - requirements 5-79
 - security 5-78
 - troubleshooting 5-98
- Cross-System Coupling A-3, A-4
- Cross-System Coupling Facility
 - CAS communication 5-76
- customization 4-1, 5-10
 - See also* AutoCustomization
 - IMS parameters 3-7
 - manual 5-1
 - procedure library 3-12
 - products 1-6
 - sample library 3-9
 - summary of 3-19
 - tasks 1-6, 4-2
 - UBBPARM parameter library 3-7
- cycle refresh C-10

D

- DAE parameter C-4
- data sets
 - configuration 3-1
 - high-level qualifier 5-2
 - historical
 - BBI-SS PAS 5-41
 - MVS PAS 5-35
 - PARMFILE
 - BBI-SS PAS 5-44
 - MVS PAS 5-38
 - product support 3-2
 - security 5-13

- site 3-13
- user 3-14
- DB2@MV MR 5-71
- DB2@RXnn 5-71
- DEBUG parameter
 - BBTTC P00 member C-9
- DESC 5-50
- dispatching priority
 - BBI-SS PAS 5-59
- distributed libraries
 - help text 3-13
 - screen definitions 3-13
 - views 3-13
- DLOGJCL 5-65
- domain 5-81
- DUMP keyword
 - CAS startup procedure A-3
- DUMPCLAS parameter C-4, C-10
- DUMPS parameter C-4, C-10
- duplicate symptom strings
 - dumps C-4
- dynamic targets 5-46, 5-83

E

- EMTMPW 5-111
- event management utility
 - AutoOPERATOR 4-5
- EXCP E-1
- EXEC
 - AutoOPERATOR 5-62
 - MAINVIEW AutoOPERATOR
 - procedure library 3-11
- Explorer, *see* MAINVIEW Explorer
- extended diagnostic mode
 - CAS startup procedure A-4
 - PAS startup procedure B-4
- extended message mode
 - CAS startup procedure A-4
- extractor
 - CMF
 - initialization B-5

G

- GDDM/PGF support 5-6, 5-19

GRAPH command 5-6
graphical user interface (GUI) 1-5
graphing support 5-6, 5-19
group
 products 4-1
GTS parameter
 BBISSPxx member C-8

H

help
 library
 description 3-4
high-level qualifier for data sets 5-2
HILITE parameter C-7
historical data set
 allocating to BBI-SS PAS 5-41
 allocating to MVS PAS 5-35
 high-level qualifier B-10
 setting up 5-35
HOLD statement 5-10
host server, MAINVIEW Explorer 5-25

I

IEAAPFxx 5-14
IEASYSxx 5-76
ILOGJCL 5-65, 5-67
image identifier 5-2
Image log
 BBI-SS PAS 3-18, 5-57
 printing 5-65
Image logs 5-57
IMAGSTC parameter C-8
IMS parameters, customizing 3-7
IMS@MVA 5-71
IMS@MVMR 5-71
IMSID 5-50
IMSRESP 5-71
IMSTYPE 5-49
initializing
 CMF extractor B-5
 MVS PAS data collectors B-6
 product 5-104
INSTall command 5-46, 5-98
installation 1-6

installation validation program 5-4
interval
 historical data 5-56
INTERVAL parameter C-3, C-10
IOCDS 5-22
IPM monitor B-7
IRRI parameter C-3
ISPF
 parameter 5-71, C-10
ISPTCM 5-111

J

JCL
 assembling a mode table entry 5-91
 CAS startup procedure 5-23
JES3CHAR parameter C-7
JES3CMD parameter C-7
JESCHAR parameter C-7
JESCNCT parameter C-4
JESFLTR parameter C-7
jobcards 5-2
Journal log
 BBI-SS PAS 3-18, 5-57
 messages C-5
 printing 5-65
JOURNAL parameter C-5
JRNLCMSG parameter C-5
JRNLCSTC parameter C-5

K

katakana terminal 5-7
 CAPS parameter 5-24, A-2
keys
 product options 5-104

L

LIBDEF 5-18
libraries
 CLISTs 3-5
 concatenated 3-16
 configuration 3-2
 distributed target 3-1
 help 3-4, 3-13

- link 3-4, 3-16
- message 3-4
- panel 3-4
- parameter 3-4
- procedure 3-4
- product user 3-2
- profile 3-4
- sample 3-4
- screen definitions 3-4
- shared 3-15
- site 3-13
 - creating 3-14
 - customized 3-1
- table 3-5
- user 3-14
- view 3-4, 3-13
- link library 3-4, 3-16
- link list 5-16, 5-20
- link list, adding load library to 5-16
- load library 5-14, 5-16
- load modules A-1, B-1
- log switching 5-65
- logging parameter C-5
- LOGON parameter C-5
- LOGPFK C-11
- LU conversations 5-88

M

MAINVIEW

- Alternate Access 5-19
 - migrating E-1
- environment
 - implementing 1-6
- introduction 1-1

MAINVIEW Alarm Manager

- BBIDIV data set 5-32
- initialization parameters 5-28
- journal logs 5-30
- PAS PROC parameters and JCL D-1
- PAS startup procedure 5-33
- product initialization 5-28, 5-29
- security requirements 5-34
- started task procedure 5-33

MAINVIEW Alternate Access 1-5

MAINVIEW AutoOPERATOR

- customized libraries 3-5

- EXEC library 3-12
- MAINVIEW CLIST 5-18
- MAINVIEW Explorer 1-5
 - BBTTCP00 parameters C-9
 - host server startup 5-25
 - TCP/IP interface C-9
- MAXDATA parameter 5-83, 5-99
- MAXPQE parameter C-5
- MAXUSER parameter C-5
- MCS console, AutoOPERATOR C-6
- MCTSIZE parameter C-7
- message library
 - description 3-4
- messages
 - BBI release 2-1
 - Journal log C-5
 - STALL DETECTED C-5
 - statistics C-7
- MIGCONS parameter C-7
- migration
 - CAS definitions 5-27
 - considerations 2-1
 - VTAM definitions 5-75
- mode table, VTAM 5-89
- MODEENT table 5-89
- MODIFY command
 - PAS B-10
- MSNF (Multisystem Networking Facility) 5-83
- multiple console support (MCS), CMDCON
 - parameter C-6
- Multisystem Networking Facility (MSNF) 5-83
- MVALARM, see MAINVIEW Alarm Manager
- MVS PAS
 - created by AutoCustomization 5-39
 - PARMFILE 5-38

N

- NCP 5-98

- NETVIEW parameter C-7

O

- online help 1-7
 - distributed 3-13
- operational defaults 5-2

option passwords 5-104

P

panel library

description 3-4

parameter library B-10

allocating A-2

AutoCustomization 3-7

description 3-4

manual customization 3-7

using 3-6

PARMFILE 3-17, 5-38, 5-44

PAS

description 1-3

password keys 5-104

PCF 5-111

PF keys 5-69, C-10, C-11

Plex Manager

description 1-3

PMGLAUTH 5-20

PRIMAPPL 5-70

Primary Option Menu panels 5-71

CICS@MR 5-71

CICS@MVA 5-71

PRIMMENU

specifying 5-70

printing

Image logs 5-65

Journal logs 5-65

PROC statement

CAS parameters A-2

MVS PAS parameters B-4

procedure library

AutoCustomization 3-12

description 3-4

manual customization 3-12

using 3-11

product

customization 1-6

group 4-1

initialization 5-104

usage 1-7

validating installation 5-4

validation program 5-4

product address space

BBI-SS

control parameters C-4

startup procedure 5-59

description 1-3

MAINVIEW Alarm Manager

startup D-1

MVS

startup B-1

startup procedure 5-39, B-1

START command 5-40

product libraries

configuration 3-2, 3-19

site-customized 3-1

types 3-1

PRODUCT parameter C-5

BBI-SS PAS 5-102

MAINVIEW Alarm Manager 5-28, 5-29

profile library

AutoCustomization 3-10

description 3-4

user 3-2

using 3-9

profile variable pool 3-17

program control facility 5-112

PROGxx 5-14

PTFs 5-10

Q

qualifier, for data sets 5-2

queue manager 5-47

R

RACF 4-5, 5-111

BBI-SS PAS 5-59

recording interval 5-56

region size

BBI-SS PAS 5-59

RELEASE 5-50

BBI 2-1

REVERSE parameter C-10

rule sets

AutoOPERATOR 4-5

RUSIZES 5-90, 5-99

S

- sample library
 - AutoCustomization 3-9
 - description 3-4
 - manual customization 3-9
 - using 3-8
- samplers B-7
- SBBPROF 5-10, 5-11
 - considerations 3-1
- SBBSDEF 5-11
- SBBSDEF screen library 3-13
- SBBVDEF 5-11
- SBBVDEF view library 3-13
- screen
 - definition
 - library 3-4, 3-13
- SCREEN parameter C-10
- security 1-7, 4-5
 - BBSECURE A-2, B-4, D-2
 - command authorization table 4-4, 5-111
 - cross-system communication
 - CAS 5-78
 - defaults 5-13
 - MAINVIEW Alarm Manager 5-34
 - required data set 5-13
- serial number, changing CPU 5-108
- site
 - data sets
 - allocating 5-10
- site libraries
 - descriptions 3-1
- SMF ID 5-36, 5-108, B-4
- SMLOAD 5-111
- SPCF CAS PROC parameter A-4
- split messages C-5
- SS0974E 5-47
- SSID
 - defining
 - BBI-SS PAS 5-49
 - CAS procedure 5-24, A-2
 - MVS PAS procedure 5-39, B-10
 - historical data sets
 - BBI-SS PAS 5-42
 - logs 5-57, 5-66
 - migration 2-5
 - PARMFILE data set
 - BBI-SS PAS 5-44
 - SSJCL 5-59, 5-110
 - SSLOG 5-57
- STACK parameter
 - BBTTCP00 member C-9
- STALL parameter C-5
- START
 - CAS A-1
 - command
 - CAS 5-24
 - MVS PAS 5-40
 - parameter 5-60
 - started task procedure
 - BBI-SS PAS 5-59
 - CAS 5-23, A-1
 - log handling 5-65
 - MAINVIEW Alarm Manager 5-33, D-1
 - MVS PAS 5-39, B-1
- startup
 - BBI-SS PAS procedure 5-59
 - CAS PROC statement A-2
 - CAS procedure 5-23, A-2
 - MAINVIEW Alarm Manager 5-33, D-1
 - MAINVIEW Explorer host server 5-25
 - MVS PAS PROC statement B-4
 - MVS PAS procedure 5-39, B-1
- subsystem 5-46
 - adding 5-80
 - communication 5-81, 5-83
- subsystem ID 5-47
 - BBI-SS PAS 5-59
 - historical data sets 5-42
 - parameter 5-49
 - PARMFILE data set 5-44
 - CAS 5-109, A-2
 - IMS control region 5-50
- SYS1.PARMLIB 5-15
 - APF authorization 5-14
 - TSF authorization 5-20
- SYS1.VTAMLIB 5-90
 - CAS definitions
 - mode table 5-100
- SYS1.VTAMLST 5-84
 - BBI-SS PAS definitions 5-85
 - CAS definitions 5-91
 - shared 5-78
- SYSID, defining 5-2
- SYSINFO 5-105
- SYSNAME 5-76

SYSOUT class B-10
sysplex communication 5-76
system parameters A-1, B-1, C-1
 ALIAS 5-50
 DESC 5-50
 IMSID 5-50
 IMSTYPE 5-49
 RELEASE 5-50
 SUBSYS 5-49
 TARGET 5-49
 TYPE 5-49

T

table library
 description of 3-5
tables A-2, B-4
TARGET
 BBIISP00 C-3
target 5-46, 5-49
 attachment 5-50
 BBITSP00 5-48
 dynamic 5-46
 jobname 5-47
 libraries 3-1
 parameters 5-46, 5-48
 recognition 5-50
 specifying 5-48, 5-69
target system parameters 5-46
tasks
 customization 1-6, 4-2
TCP/IP interface
 BBISSPxx member C-8
 BBTTCP00 member C-9
TCPNAME parameter
 BBTTCP00 member C-9
terminal session 1-1
 description 1-5
 parameters
 BBIRESP 5-71
 IMSRESP 5-71
 ISPF 5-71
 PRIMAPPL 5-70
 PRIMMENU 5-70
 TARGET 5-69
 TYPE 5-69, 5-70
UAS 1-5

TRANSfer command 5-20
TRDIR parameter C-3
TS
 See terminal session
 TSO command processor 5-111
TSF authorization 5-20
TSLOAD 5-111
TYPE 5-49, 5-69, 5-70

U

UAS
 See user address space
UBBPARM 5-10, 5-11, B-10
 considerations 3-1
UBBSAMP 5-10, 5-11
 considerations 3-1
user
 profile 3-2
user address space
 description 1-5
using a product 1-7
USRID parameter C-3

V

validating product installation 5-4
view B-4
 libraries
 distributed 3-13, A-2, B-4
 table A-2
views
 BBVDEF 3-13
 distributed libraries 3-13
 library containing product 3-4
 SBBVDEF library 3-13
virtual storage requirements 5-3
VTAM
 BBI-SS PAS definition 5-83
CAS
 activating major nodes 5-93
 definition 5-75, 5-87
 updating the configuration list 5-92
defining
 migration 5-75
MAXDATA 5-83

mode table 5-89
resource manager
 BBI-SS PAS definitions 5-84
 CAS definitions 5-91
subsystem communication 5-81
terminal session
 MAINVIEW Alternate Access E-1

W

WARM start BBI-SS PAS 5-60
Web browser 1-5

X

XCF
 communication link 5-75
 customizing group name 5-94
XCF, enabling for CAS A-3, A-4
XCFGROUP parameter C-7
XDS B-8
X-MCS consoles C-7

STOP!

IMPORTANT INFORMATION - DO NOT INSTALL THIS PRODUCT UNLESS YOU HAVE READ ALL OF THE FOLLOWING MATERIAL

By clicking the YES or ACCEPT button below (when applicable), or by installing and using this Product or by having it installed and used on your behalf, You are taking affirmative action to signify that You are entering into a legal agreement and are agreeing to be bound by its terms, EVEN WITHOUT YOUR SIGNATURE. BMC is willing to license this Product to You ONLY if You are willing to accept all of these terms. CAREFULLY READ THIS AGREEMENT. If You DO NOT AGREE with its terms, DO NOT install or use this Product; press the NO or REJECT button below (when applicable) or promptly contact BMC or your BMC reseller and your money will be refunded if by such time You have already purchased a full-use License.

SOFTWARE LICENSE AGREEMENT FOR BMC PRODUCTS

SCOPE. This is a legally binding Software License Agreement (“License”) between You (either an individual or an entity) and BMC pertaining to the original computer files (including all computer programs and data stored in such files) contained in the enclosed Media (as defined below) or made accessible to You for electronic delivery, if as a prerequisite to such accessibility You are required to indicate your acceptance of the terms of this License, and all whole or partial copies thereof, including modified copies and portions merged into other programs (collectively, the “Software”). “Documentation” means the related hard-copy or electronically reproducible technical documents furnished in association with the Software, “Media” means the original BMC-supplied physical materials (if any) containing the Software and/or Documentation, “Product” means collectively the Media, Software, and Documentation, and all Product updates subsequently provided to You, and “You” means the owner or lessee of the hardware on which the Software is installed and/or used. “BMC” means BMC Software Distribution, Inc. unless You are located in one of the following regions, in which case “BMC” refers to the following indicated BMC Software, Inc. subsidiary: (i) Europe, Middle East or Africa --BMC Software Distribution, B.V., (ii) Asia/Pacific -- BMC Software Asia Pacific Pte Ltd., (iii) Brazil -- BMC Software do Brazil, or (iv) Japan -- BMC Software K.K. If You enter into a separate, written software license agreement signed by both You and BMC or your authorized BMC reseller granting to you the rights to install and use this Product, then the terms of that separate, signed agreement will apply and this License is void.

FULL-USE LICENSE. Subject to these terms and payment of the applicable license fees, BMC grants You this non-exclusive License to install and use one copy of the Software for your internal use on the number(s) and type(s) of servers or workstations for which You have paid or agreed to pay to BMC or your BMC reseller the appropriate license fee. If your license fee entitles You only to a License having a limited term, then the duration of this License is limited to that term; otherwise this License is perpetual, subject to the termination provisions below.

TRIAL LICENSE. If You have not paid or agreed to pay to BMC or your BMC Reseller the appropriate license fees for a full use license, then, **NOTWITHSTANDING ANYTHING TO THE CONTRARY CONTAINED IN THIS LICENSE:** (i) this License consists of a non-exclusive evaluation license (“Trial License”) to use the Product for a limited time (“Trial Period”) only for evaluation; (ii) during the Trial Period, You may not use the Software for development, commercial, production, database management or other purposes than those expressly permitted in clause (i) immediately above; and (iii) your use of the Product is on an **AS IS** basis, and **BMC, ITS RESELLERS AND LICENSORS GRANT NO WARRANTIES OR CONDITIONS (INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) TO YOU AND ACCEPT NO LIABILITY WHATSOEVER RESULTING FROM THE USE OF THIS PRODUCT UNDER THIS TRIAL LICENSE.** If You use this Product for other than evaluation purposes or wish to continue using it after the Trial Period, you must purchase a full-use license. When the Trial Period ends, your right to use this Product automatically expires, though in certain cases You may be able to extend the term of the Trial Period by request. Contact BMC or your BMC reseller for details.

TERM AND TERMINATION. This License takes effect on the first to occur of the date of shipment or accessibility to You for electronic delivery, as applicable (the “Product Effective Date”). You may terminate this License at any time for any reason by written notice to BMC or your BMC reseller. This License and your right to use the Product will terminate automatically with or without notice by BMC if You fail to comply with any material term of this License. Upon termination, You must erase or destroy all components of the Product including all copies of the Software, and stop using or accessing the Software. Provisions concerning Title and Copyright, Restrictions (or Restricted Rights, if You are a U.S. Government entity) or limiting BMC’s liability or responsibility shall survive any such termination.

TITLE AND COPYRIGHT; RESTRICTIONS. All title and copyrights in and to the Product, including but not limited to all modifications thereto, are owned by BMC and/or its affiliates and licensors, and are protected by both United States copyright law and applicable international copyright treaties. You will not claim or assert title to or ownership of the Product. To the extent expressly permitted by applicable law or treaty notwithstanding this limitation, You may copy the Software only for backup or archival purposes, or as an essential step in utilizing the Software, but for no other purpose. You will not remove or alter any copyright or proprietary notice from copies of the Product. You acknowledge that the Product contains valuable trade secrets of BMC and/or its affiliates and licensors. Except in accordance with the terms of this License, You agree (a) not to decompile, disassemble, reverse engineer or otherwise attempt to derive the Software’s source code from object code except to the extent expressly permitted by applicable law or treaty despite this limitation; (b) not to sell, rent, lease, license, sublicense, display, modify, time share, outsource or otherwise transfer the Product to, or permit the use of this Product by, any third party; and (c) to use reasonable care and protection to prevent the unauthorized use, copying, publication or dissemination of the Product and BMC confidential information learned from your use of the Product. **You will not export or re-export any Product without both the written consent of BMC and the appropriate U.S. and/ or foreign government license(s) or license exception(s).** Any programs, utilities, modules or other software or documentation created, developed, modified or enhanced by or for You using this Product shall likewise be subject to these restrictions. BMC has the right to obtain injunctive relief against any actual or threatened violation of these restrictions, in addition to any other available remedies. Additional restrictions may apply to certain files, programs or data supplied by third parties and embedded in the Product; consult the Product installation instructions or Release Notes for details.

LIMITED WARRANTY AND CONDITION. If You have purchased a Full-Use License, BMC warrants that (i) the Media will be, under normal use, free from physical defects, and (ii) for a period of ninety (90) days from the Product Effective Date, the Product will perform in substantial accordance with the operating specifications contained in the Documentation that is most current at the Product Effective Date. BMC’s entire liability and your exclusive remedy under this provision will be for BMC to use reasonable best efforts to remedy defects covered by this warranty

and condition within a reasonable period of time or, at BMC's option, either to replace the defective Product or to refund the amount paid by You to license the use of the Product. BMC and its suppliers do not warrant that the Product will satisfy your requirements, that the operation of the Product will be uninterrupted or error free, or that all software defects can be corrected. This warranty and condition shall not apply if: (i) the Product is not used in accordance with BMC's instructions, (ii) a Product defect has been caused by any of your or a third party's malfunctioning equipment, (iii) any other cause within your control causes the Product to malfunction, or (iv) You have made modifications to the Product not expressly authorized in writing by BMC. No employee, agent or representative of BMC has authority to bind BMC to any oral representations, warranties or conditions concerning the Product. **THIS WARRANTY AND CONDITION IS IN LIEU OF ALL OTHER WARRANTIES AND CONDITIONS. THERE ARE NO OTHER EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS, INCLUDING THOSE OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, REGARDING THIS LICENSE OR ANY PRODUCT LICENSED HEREUNDER. THIS PARAGRAPH SHALL NOT APPLY TO A TRIAL LICENSE.** Additional support and maintenance may be available for an additional charge; contact BMC or your BMC reseller for details.

LIMITATION OF LIABILITY. Except as stated in the next succeeding paragraph, BMC's and your BMC reseller's total liability for all damages in connection with this License is limited to the price paid for the License. **IN NO EVENT SHALL BMC BE LIABLE FOR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, PUNITIVE OR INDIRECT DAMAGES OF ANY KIND ARISING OUT OF THE USE OF THIS PRODUCT (SUCH AS LOSS OF PROFITS, GOODWILL, BUSINESS, DATA OR COMPUTER TIME, OR THE COSTS OF RECREATING LOST DATA), EVEN IF BMC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Some jurisdictions do not permit the limitation of consequential damages so the above limitation may not apply.

INDEMNIFICATION FOR INFRINGEMENT. BMC will defend or settle, at its own expense, any claim against You by a third party asserting that your use of the Product within the scope of this License violates such third party's patent, copyright, trademark, trade secret or other proprietary rights, and will indemnify You against any damages finally awarded against You arising out of such claim. However, You must promptly notify BMC in writing after first receiving notice of any such claim, and BMC will have sole control of the defense of any action and all negotiations for its settlement or compromise, with your reasonable assistance. BMC will not be liable for any costs or expenditures incurred by You without BMC's prior written consent. If an order is obtained against your use of the Product by reason of any claimed infringement, or if in BMC's opinion the Product is likely to become the subject of such a claim, BMC will at its option and expense either (i) procure for You the right to continue using the product, or (ii) modify or replace the Product with a compatible, functionally equivalent, non-infringing Product, or (iii) if neither (i) nor (ii) is practicable, issue to You a pro-rata refund of your paid license fee(s) proportionate to the number of months remaining in the 36 month period following the Product Effective Date. This paragraph sets forth your only remedies and the total liability to You of BMC, its resellers and licensors arising out of such claims.

GENERAL. This License is the entire understanding between You and BMC concerning this License and may be modified only in a mutually signed writing between You and BMC. If any part of it is invalid or unenforceable, that part will be construed, limited, modified, or severed so as to eliminate its invalidity or unenforceability. This License will be governed by and interpreted under the laws of the jurisdiction named below, without regard to conflicts of law principles, depending on which BMC Software, Inc. subsidiary is the party to this License: (i) BMC Software Distribution, Inc. - the State of Texas, U.S.A., (ii) BMC Software Distribution, B.V. - The Netherlands, (iii) BMC Software Asia Pacific Pte Ltd. -- Singapore (iv) BMC Software do Brazil -- Brazil, or (v) BMC Software K.K. -- Japan. Any person who accepts or signs changes to the terms of this License promises that they have read and understood these terms, that they have the authority to accept on your behalf and legally obligate You to this License. Under local law and treaties, the restrictions and limitations of this License may not apply to You; You may have other rights and remedies, and be subject to other restrictions and limitations.

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b) or DFARS 227.7202, 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.

Notes



17882