

# **MAINVIEW<sup>®</sup> AutoOPERATOR**

## **Customization Guide**

**Version 6.3.01**

**November 21, 2003**



Copyright 2003 BMC Software, Inc. 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. IBM and DB2 are registered trademarks of International Business Machines Corp. All other trademarks belong to their respective companies.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this documentation.

## Restricted Rights Legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

---

## 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](http://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

<b>Chapter 1</b>	<b>Migrating from MAINVIEW AutoOPERATOR Version 6.3.00 to Version 6.3.01</b>	
	What You Need to Know . . . . .	1-1
	New Product Option . . . . .	1-2
	Changes Made to Existing Parameters, Rules, and EXECs . . . . .	1-2
	SAP High Availability Issues . . . . .	1-2
	Total Object Manager Issues . . . . .	1-3
	MAINVIEW AutoOPERATOR 6.3.01 Overhead Reduction . . . . .	1-3
	General Questions and Answers about the New Release . . . . .	1-4
	What Has Changed about Rules . . . . .	1-4
	What DD Names Have Changed . . . . .	1-4
	What Parameters Have Changed . . . . .	1-4
	What Are the Changes Related to IMS . . . . .	1-4
	What Are the Changes Related to CICS . . . . .	1-4
	What Else Has Changed . . . . .	1-5
	How to Secure MAINVIEW AutoOPERATOR 6.3.01 . . . . .	1-5
	How Much CPU Does the New Release Use . . . . .	1-5
	Does MAINVIEW AutoOPERATOR 6.3.01 Use Common Storage Differently . . . . .	1-5
	Does MAINVIEW AutoOPERATOR 6.3.01 Use Private Storage Differently . . . . .	1-5
	Does This Version Communicate with Version 6.3.00 . . . . .	1-6
	Does Version 6.2 Communicate with This Version . . . . .	1-6
	Which IBM Software Does MAINVIEW AutoOPERATOR 6.3.01 Require . . . . .	1-6
	What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01 . . . . .	1-6
	What Is Not Supported by Version 6.3.01 . . . . .	1-7
	What Functions Will Not Be Supported in the Next Release of MAINVIEW AutoOPERATOR . . . . .	1-7
	What Functions Will Not Be Supported in Future Releases of MAINVIEW AutoOPERATOR . . . . .	1-7

---

Installation Considerations for MAINVIEW AutoOPERATOR	
6.3.01	1-8
Getting New BBKEYS	1-8
Defining MAINVIEW AutoOPERATOR Resources to CICS	1-8
Migrating by Using the OS/390 and z/OS Installer: Standard	
Installation	1-8
AutoCustomization Considerations When Replacing Existing	
Products	1-9
Migrating by Using System Maintenance Program (SMP)	1-10

## Chapter 2

### Migrating from MAINVIEW AutoOPERATOR Version 6.2.00 to Version 6.3.01

What You Need to Know	2-1
New Product Option	2-2
Changes Made to Existing Parameters, Rules, and EXECs	2-2
CSM Compatibility Issues	2-3
MAINVIEW AutoOPERATOR 6.3.01 Overhead Reduction	2-3
MAINVIEW AutoOPERATOR and PATROL Enterprise	
Manager Connections	2-3
General Questions and Answers about the New Release	2-4
What Has Changed about Rules	2-4
What Are the Changes Made to Variables	2-5
What DD Names Have Changed	2-5
What Parameters Have Changed	2-5
What Are the Changes Related to IMS	2-6
What Are the Changes Related to CICS	2-6
What Else Has Changed	2-6
How to Secure MAINVIEW AutoOPERATOR 6.3.01	2-6
How Much CPU Does the New Release Use	2-7
Does MAINVIEW AutoOPERATOR 6.3.01 Use Common	
Storage Differently	2-7
Does MAINVIEW AutoOPERATOR 6.3.01 Use Private	
Storage Differently	2-7
Does This Version Communicate with Version 6.2.00	2-7
Does Version 6.2.00 Communicate with This Version	2-7
Which IBM Software Does MAINVIEW AutoOPERATOR	
6.3.01 Require	2-8
What Other BMC Software Products Run with MAINVIEW	
AutoOPERATOR 6.3.01	2-8
What Is Not Supported by Version 6.3.01	2-8
What Functions Will Not Be Supported in the Next Release of	
MAINVIEW AutoOPERATOR	2-9
What Functions Will Not Be Supported in Future Releases of	
MAINVIEW AutoOPERATOR	2-9

Installation Considerations for MAINVIEW AutoOPERATOR	
6.3.01	2-10
Getting New BBKEYS	2-10
Defining MAINVIEW AutoOPERATOR Resources to CICS	2-10
Migrating by Using the OS/390 and z/OS Installer: Standard	
Installation	2-10
AutoCustomization Considerations When Replacing Existing	
Products	2-11
Migrating by Using System Maintenance Program (SMP)	2-12

## Chapter 3

### Migrating from MAINVIEW AutoOPERATOR Version 6.1.00 to Version 6.3.01

What You Need to Know	3-1
New Product Option	3-2
Messages That Have Changed	3-2
Changes Made to Existing Parameters, Rules, and EXECs	3-3
CSM Compatibility Issues	3-3
MAINVIEW AutoOPERATOR 6.3.01 Overhead Reduction	3-4
MAINVIEW AutoOPERATOR and PATROL Enterprise	
Manager Connections	3-4
General Questions and Answers about the New Release	3-5
What Has Changed about Rules	3-5
What Are the Changes Made to Variables	3-7
What DD Names Have Changed	3-7
What Parameters Have Been Changed or Added	3-8
What Are the Changes Related to IMS	3-9
What Are the Changes Related to CICS	3-9
What Else Has Changed	3-9
How to Secure MAINVIEW AutoOPERATOR 6.3.01	3-10
How Much CPU Does the New Release Use	3-10
Does MAINVIEW AutoOPERATOR 6.3.01 Use Common	
Storage Differently	3-11
Does MAINVIEW AutoOPERATOR 6.3.01 Use Private	
Storage Differently	3-11
Does This Version Communicate with Version 6.1.00	3-11
Does Version 6.1.00 Communicate with This Version	3-11
Which IBM Software Does MAINVIEW AutoOPERATOR	
6.3.01 Require	3-12
What Other BMC Software Products Run with MAINVIEW	
AutoOPERATOR 6.3.01	3-12
What Is Not Supported by Version 6.3.01	3-12
What Functions Will Not Be Supported in the Next Release of	
MAINVIEW AutoOPERATOR	3-13
What Functions Will Not Be Supported in Future Releases of	
MAINVIEW AutoOPERATOR	3-13

Installation Considerations for MAINVIEW AutoOPERATOR	
6.3.01	3-14
Getting New BBKEYS	3-14
Defining MAINVIEW AutoOPERATOR Resources to CICS	3-14
Migrating by Using the OS/390 and z/OS Installer: Standard	
Installation	3-14
AutoCustomization Considerations When Replacing Existing	
Products	3-15
Migrating by Using System Maintenance Program (SMP)	3-16

## Chapter 4

### Continuous Operation: Using the Dynamic Parameter Manager

Tasks That You Can Accomplish with the Dynamic Parameter	
Manager	4-1
Accessing the Dynamic Parameter Manager	4-2
Describing the Fields	4-6
Filtering the Display of Data	4-6
Modifying ALERT Thresholds in AAOALSxx	4-7
Setting ALERTs Storage Thresholds	4-9
Modifying Automation Reporter Application Parameters in	
AAOARPxx	4-13
Setting Automation Reporter Parameters	4-15
Modifying EXEC Parameters in AAOEXPxx	4-17
Setting EXEC Thresholds	4-19
Setting EXEC Parameters	4-25
Setting High-Priority EXECs	4-28
Modifying TapeSHARE for MAINVIEW AutoOPERATOR	
Parameters in AAOTSPxx	4-30
Customizing TapeSHARE	4-32
Dynamically Implementing New Parameter Settings with the Activate	
Command	4-38

## Chapter 5

### MVS Console Considerations

Determining How Many Consoles to Allocate	5-2
Extended MCS (X-MCS) Consoles	5-2
Allocating X-MCS Consoles without MIGIDs	5-3
Allocating X-MCS Consoles with MIGIDs	5-3
Defining X-MCS Consoles to CICS	5-4

## Chapter 6

### Implementing TapeSHARE

Overview	6-1
Prerequisites	6-1
Step 1: Entering the Password Key for TapeSHARE (Required)	6-2
Step 2: Activating TapeSHARE (Required)	6-2
Step 3: Securing TapeSHARE (Optional)	6-2
After TapeSHARE Is Installed	6-3

<b>Chapter 7</b>	<b>Implementing Automatic Restart Manager (ARM)</b>	
	Overview . . . . .	7-1
	Implementation Considerations . . . . .	7-2
	Implementing ARM Support: Default ARM Policy . . . . .	7-2
	Implementing ARM Support: Creating a User-Defined Policy . . . . .	7-3
	Enabling ARM Support . . . . .	7-3
	Stopping MAINVIEW AutoOPERATOR When ARM Is Enabled . . . . .	7-4
<b>Chapter 8</b>	<b>Implementing Support for MPF</b>	
	Overview . . . . .	8-1
	Implementation Considerations . . . . .	8-2
	Setting the MPFFILTR Parameter . . . . .	8-3
<b>Chapter 9</b>	<b>Activating the REXX/370 Alternate Library</b>	
	SMP/E Messages . . . . .	9-2
<b>Chapter 10</b>	<b>Defining OSPI Virtual Terminals</b>	
	Overview . . . . .	10-1
	Using the OSPI ACB Pool . . . . .	10-2
	Naming Conventions . . . . .	10-2
	Defining the ACB Pool . . . . .	10-3
	Defining the ACB Pool to MAINVIEW AutoOPERATOR . . . . .	10-3
	OSPI Application Definitions . . . . .	10-4
<b>Chapter 11</b>	<b>Implementing BBI-SS PAS to CICS Communication</b>	
	Overview . . . . .	11-1
	Implementation Considerations . . . . .	11-2
	Site-Specific Security . . . . .	11-2
	Temporary Storage Table (TST) Compatibility . . . . .	11-2
	Specifying Resources to CICS . . . . .	11-3
	Making the MAINVIEW AutoOPERATOR Transactions and Programs Available to CICS/ESA and CICS Transaction Server . . . . .	11-3
	Modifying the Program List Table (PLT) . . . . .	11-3
	Modifying CICS Startup JCL . . . . .	11-5
	Restarting CICS Target Regions . . . . .	11-6
	MAINVIEW AutoOPERATOR for CICS Functions (with BBI-SS PAS-to-CICS Communication) . . . . .	11-7
	MAINVIEW AutoOPERATOR for CICS Functions (without BBI-SS PAS-to-CICS Communication) . . . . .	11-9
<b>Chapter 12</b>	<b>MAINVIEW AutoOPERATOR for CICS Optional Tailoring</b>	
	Overview . . . . .	12-1
	Customizing Terminal Types Eligible for Broadcast Messages (CAOTTAB) . . . . .	12-2
	Limiting CICS Transient Data Queues Eligible for Rule Processing (CAODTAB) . . . . .	12-3

	Using the CAODTAB Macro . . . . .	12-4
	Describing CAODTAB Macro Parameters . . . . .	12-4
<b>Chapter 13</b>	<b>Implementing MAINVIEW Products in IMS</b>	
	Deleting Modules from a Prior Release . . . . .	13-2
	Setting Up BBPARM Data Sets . . . . .	13-3
	Customizing BBPARM Members for an IMS System . . . . .	13-4
	Activating MAINVIEW AutoOPERATOR for IMS . . . . .	13-5
	Enabling Product Initialization . . . . .	13-5
	Enabling Product Authorization . . . . .	13-5
	Modifying the IMS Control Region JCL . . . . .	13-6
	Allocating a BBPARM Data Set . . . . .	13-6
	Giving IMS Access to MAINVIEW AutoOPERATOR for IMS . . . . .	13-6
	Enabling AO Exit Routines . . . . .	13-7
	Stopping Message Queuing to the IMS Master Terminal . . . . .	13-11
<b>Chapter 14</b>	<b>Implementing MAINVIEW AutoOPERATOR Access NV</b>	
	Overview . . . . .	14-2
	Modifying NetView JCL . . . . .	14-3
	Creating DSIPARM Member NAIOPT00 . . . . .	14-4
	Modifying DSIPARM Member DSICMD . . . . .	14-4
	Modifying DSIPARM Member DSIDMN . . . . .	14-5
	Copying BBSAMP Members to the NetView DSICLD Data Set . . . . .	14-6
	Modifying DSIPARM Member DSIOPF . . . . .	14-6
	Verifying Operator Profiles . . . . .	14-6
	Reviewing NetView Security for Access NV Operator IDs . . . . .	14-7
	Reviewing the Use of NetView Exit DSIEX02A . . . . .	14-8
	Reviewing the Use of NetView Exit DSIEX14 . . . . .	14-8
	Modifying BBPARM Member BBISSP00 . . . . .	14-9
<b>Chapter 15</b>	<b>Implementing AutoRULE</b>	
	Overview . . . . .	15-1
	Prerequisites . . . . .	15-1
	Completing the AutoRULE AutoCustomization Step . . . . .	15-2
<b>Chapter 16</b>	<b>Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM</b>	
	Overview . . . . .	16-1
	Establishing a TCP/IP Connection . . . . .	16-4
	What You Need . . . . .	16-4
	Defining TCP/IP Addresses in MAINVIEW AutoOPERATOR . . . . .	16-6
	Resetting BBPARM Member AAOGME00 . . . . .	16-7
	Controlling the MAINVIEW AutoOPERATOR and PATROL Enterprise Manager Interface . . . . .	16-7
	Modifying the TCP/IP Connection with BBPARM Member AAOGME00 . . . . .	16-9

---

	How MAINVIEW AutoOPERATOR and PATROL EM Share ALERTs . . . . .	16-10
	ALERTs Synchronization . . . . .	16-10
	Creating and Deleting ALERT and ALERT Queues . . . . .	16-11
<b>Chapter 17</b>	<b>Implementing MAINVIEW AutoOPERATOR Advanced JES3 Support</b>	
	Using JES3 2.2 and Later . . . . .	17-1
	Intercepting Messages . . . . .	17-2
	Intercepting Commands . . . . .	17-2
	Issuing JES3 Commands from MAINVIEW AutoOPERATOR . . . . .	17-4
	Using JES3 Version Prior to 2.2 . . . . .	17-4
	Intercepting Messages . . . . .	17-5
	Intercepting Commands . . . . .	17-5
	Issuing JES3 Commands from MAINVIEW AutoOPERATOR . . . . .	17-5
<b>Chapter 18</b>	<b>Implementing Continuous State Manager</b>	
	Performance Considerations . . . . .	18-1
	Initiating CSM . . . . .	18-1
	Overriding the Default Group when CSM Is Initiated . . . . .	18-2
	End-of-Memory Initiated CSM EXEC (CSMEOM) . . . . .	18-3
	Accessing CSM for the First Time . . . . .	18-3
	Migrating from a Previous Version of MAINVIEW AutoOPERATOR . . . . .	18-6
	Creating a Shared Repository Data Set for CSM . . . . .	18-7
	Migrating a CSM Database from a Previous Release . . . . .	18-9
	Resolving Object-Definition Conflicts after Converting . . . . .	18-15
<b>Chapter 19</b>	<b>Specific MAINVIEW AutoOPERATOR Parameters and JCL</b>	
	Enabling REXX Support for EXECs . . . . .	19-1
	Detecting EXEC Loops . . . . .	19-2
	Prioritizing and Threading of EXECs . . . . .	19-2
	Defining the Message Counter Table . . . . .	19-3
	Displaying MVS JES2 Job Numbers in Address Space Application . . . . .	19-4
	Starting BBI-SS PAS before JES . . . . .	19-5
	IMFJNUM Option and Seven-Digit JES Job Numbers (z/OS v1.2 and Later) . . . . .	19-6
<b>Chapter 20</b>	<b>Accessing Advanced SYSPROG Services from MAINVIEW AutoOPERATOR</b>	
	Executing AEWS in the MAINVIEW AutoOPERATOR Address Space . . . . .	20-2
<b>Chapter 21</b>	<b>Authorization for Special MAINVIEW AutoOPERATOR Functions</b>	
	Command Authorization . . . . .	21-1

<b>Chapter 22</b>	<b>Authorizing EXECs Scheduled by an Operator or the Rule Processor</b>	
<b>Chapter 23</b>	<b>Implementing MAINVIEW AutoOPERATOR for CICS Function Security (CMRSECU)</b>	
	CMRSECU TYPE=INITIAL Statement (Required) . . . . .	23-2
	CMRSECU TYPE=ENTRY Statement by Operator ID . . . . .	23-7
	CMRSECU TYPE=FINAL Statement (Required) . . . . .	23-8
<b>Appendix A</b>	<b>Issuing Commands from Terminal Sessions</b>	
	Issuing OS/390 Commands . . . . .	A-1
	Using Consoles to Issue OS/390 Commands . . . . .	A-2
	Issuing BBI Commands . . . . .	A-2
	Issuing IMS Commands . . . . .	A-3
	Issuing DBCTL Commands . . . . .	A-3
	Issuing DBCTL Commands from a Terminal Session . . . . .	A-3
	If You Have Authority to Issue MVS Commands . . . . .	A-3
	If You Have Authority to Issue IMS Commands . . . . .	A-4
<b>Appendix B</b>	<b>Changing the Use of CICS Recoverable Temporary Storage</b>	
<b>Appendix C</b>	<b>CICS Transaction and Program Names</b>	
	Transaction Entries for MAINVIEW for CICS and MAINVIEW	
	AutoOPERATOR for CICS . . . . .	C-1
	Program Entries for MAINVIEW for CICS and MAINVIEW	
	AutoOPERATOR for CICS . . . . .	C-3
	Release-Independent Programs . . . . .	C-3
	Release-Dependent Programs . . . . .	C-3
<b>Appendix D</b>	<b>Manual Activation of BBI-SS to CICS Communication</b>	
	Manual Activation from a CICS Terminal . . . . .	D-1
	Manual Activation from an OS/390 Console that Is Defined to CICS . . . . .	D-2
<b>Appendix E</b>	<b>BBPARM Data Set Members for MAINVIEW AutoOPERATOR</b>	
	System Variables Substitution in Parameter Library Members . . . . .	E-4
	BBPARM Member AAOALS00 Parameters	
	(MAINVIEW AutoOPERATOR Only) . . . . .	E-6
	BBPARM Member AAOALT00 Parameters (MAINVIEW AutoOPERATOR Only) . . . . .	E-8
	BBPARM Member AAOARP00 Parameters (MAINVIEW AutoOPERATOR Only) . . . . .	E-10
	BBPARM Member AAOEWD00 Parameters (MAINVIEW AutoOPERATOR Only) . . . . .	E-12
	BBPARM Member AAOEXP00 Parameters (MAINVIEW AutoOPERATOR Only) . . . . .	E-13
	BBPARM Member AAOGME00 Parameters (MAINVIEW AutoOPERATOR Only) . . . . .	E-16

	BBPARM Member AAOMQL00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-20
	BBPARM Member AAONCD00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-23
	BBPARM Member AAOPRM00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-24
	BBPARM Members AAORUL00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-27
	BBPARM Member AAOTRN00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-29
	BBPARM Member AAOTSP00 Parameters (MAINVIEW AutoOPERATOR Only) .....	E-30
	BBPARM Member BBIVAR00 Parameters (MAINVIEW AutoOPERATOR Only).....	E-38
<b>Appendix F</b>	<b>Sample Data Set Members</b>	
	Sample Members for MAINVIEW AutoOPERATOR Only .....	F-1
	Sample Members for MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS .....	F-3
	Sample Members for MAINVIEW AutoOPERATOR for IMS, MAINVIEW for IMS, and MAINVIEW for DBCTL.....	F-4
<b>Appendix G</b>	<b>Expanding the PROFILE Pool Data Set</b>	
	Using the XTLOAD Utility .....	G-1
<b>Appendix H</b>	<b>AutoRULE Worksheet</b>	
	Section 1: MPF List Conversion Parameters .....	H-1
	Section 2: AutoMate Conversion Parameters .....	H-3
	Section 3: Generating Rules .....	H-3
	Section 4: Creating Rules for Production Job Abends .....	H-5
	Section 5: Creating Rules for Production Job Abends .....	H-5
	Describing AutoRULE Flags .....	H-7
	Selection Flags .....	H-7
	Action Flags .....	H-8
	Special Flags .....	H-9
<b>Appendix I</b>	<b>How Product Libraries Should Be Used</b>	
	Parameter Library .....	I-3
	BBPARM .....	I-3
	UBBPARM.....	I-4
	IMS-Specific Parameter Library (IMF and MAINVIEW AutoOPERATOR for IMS Only) .....	I-4
	Sample Library .....	I-5
	BBSAMP .....	I-5
	UBBSAMP.....	I-5

Profile Library .....	I-6
BBPROF .....	I-7
SBBPROF .....	I-7
User BBPROF .....	I-8
Procedure Library (MAINVIEW AutoOPERATOR Only) .....	I-8
BBPROC .....	I-8
UBBPROC .....	I-8
BBI-2 Product Library Use Summary .....	I-9

## Appendix J

### OSPI Terminal Definitions for CICS and IMS

IMS Terminal Definitions .....	J-1
CICS Terminal Definitions .....	J-2
Autoinstall Terminals .....	J-2
Explicit Terminal Definitions .....	J-2
RDO Macro .....	J-3

## Appendix K

### Establishing a VTAM Connection for MAINVIEW AutoOPERATOR to PATROL EM Communication

Defining VTAM ACBs .....	K-2
Defining 3270 Printers to VTAM .....	K-2
Resetting BBPARM Member AAOALT00 .....	K-3
Controlling the MAINVIEW AutoOPERATOR to PATROL EM Interface .....	K-4
Modifying the VTAM Connection with BBPARM Member AAOALT00 .....	K-5
3287 Data Stream Format for MAINVIEW AutoOPERATOR to PATROL EM .....	K-6
Diagnostics .....	K-8

## Index

---

---

# Figures

Figure 4-1	MAINVIEW AutoOPERATOR PRIMARY OPTION MENU . . . . .	4-3
Figure 4-2	Dynamic Parameter Manager—Example 1 . . . . .	4-4
Figure 4-3	Dynamic Parameter Manager Panel—Example 2 . . . . .	4-7
Figure 4-4	ALERTs Storage Thresholds Panel . . . . .	4-9
Figure 4-5	Dynamic Parameter Manager Panel—Example 3 . . . . .	4-13
Figure 4-6	Automation Reporter Parm's Panel . . . . .	4-15
Figure 4-7	Dynamic Parameter Manager Member Panel . . . . .	4-18
Figure 4-8	EXEC Thresholds Panel . . . . .	4-19
Figure 4-9	EXEC Parameters Panel . . . . .	4-25
Figure 4-10	Update Non-Unique Parameter List . . . . .	4-29
Figure 4-11	Dynamic Parameter Manager Panel—Example 4 . . . . .	4-30
Figure 4-12	TapeSHARE Parameters Panel . . . . .	4-32
Figure 4-13	Confirm Parameter Activation Panel . . . . .	4-39
Figure 15-1	AutoRULE Event Management: Panel 1 . . . . .	15-2
Figure 15-2	AutoRULE Event Management: Panel 2 . . . . .	15-3
Figure 15-3	AutoRULE Event Management: Compare Option . . . . .	15-4
Figure 15-4	AutoRULE Event Management: Compare Option Completed . . . . .	15-5
Figure 15-5	AutoRULE Event Management: Naming Rule Sets . . . . .	15-6
Figure 15-6	AutoRULE Event Management: Creating a Production Job . . . . .	15-7
Figure 15-7	AutoRULE Event Management: Specifying ALERT Queues . . . . .	15-8
Figure 15-8	AutoRULE Event Management: Successful Completion Panel . . . . .	15-9
Figure 16-1	MAINVIEW AutoOPERATOR and PATROL EM Connection Using TCP/IP . . . . .	16-2
Figure 16-2	MAINVIEW AutoOPERATOR and PATROL EM Connections through I/O Concepts X-Direct Protocol Converters . . . . .	16-3
Figure 18-1	Example of Offloaded Data . . . . .	18-6
Figure 18-2	BBSAMP member INITCSM Sample JCL . . . . .	18-8
Figure 18-3	Calculating Repository Size . . . . .	18-9
Figure 18-4	CSM Global Overview Panel . . . . .	18-10
Figure 18-5	CSM Conversion Panel . . . . .	18-11
Figure 18-6	Completed Group Detail Control Panel . . . . .	18-12
Figure 18-7	Pop-Up Window for New Group Name . . . . .	18-13
Figure 18-8	CSM Conversion Status Pop-Up Window . . . . .	18-13
Figure 18-9	CSM Conversion Log . . . . .	18-14

---

Figure 18-10	Completed CSM Data Set Conversion . . . . .	18-15
Figure 18-11	JCL Example for Copying Shared Repository . . . . .	18-17
Figure G-1	Sample JCL to Execute XTLOAD . . . . .	G-2
Figure I-1	BBI-2 Data Set Customization . . . . .	I-2
Figure J-1	Example of Using Resource Definition Online (Part 1 of 2) . . . . .	J-3
Figure J-2	Example of Using Resource Definition Online (Part 2 of 2) . . . . .	J-4

---

---

# Tables

Table 11-1	Syntax for the CMRPLT Macro . . . . .	11-4
Table 13-1	BBSAMP Jobs to Delete Old Modules . . . . .	13-2
Table 16-1	BBI Control Commands for MAINVIEW AutoOPERATOR and PATROL EM Interface for TCP/IP Connections . . . . .	16-8
Table 18-1	Object Definitions . . . . .	18-16
Table 23-1	Action Parameters for the CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements . . . . .	23-3
Table 23-2	Display Parameters for CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements . . . . .	23-5
Table C-1	Transaction IDs . . . . .	C-1
Table C-2	ZAPs for Transaction ID Changes . . . . .	C-2
Table C-3	Release-Independent Programs . . . . .	C-3
Table C-4	Release-Dependent Programs . . . . .	C-4
Table E-1	BBPARM Data Set Members: MAINVIEW AutoOPERATOR Only . . . . .	E-1
Table E-2	PROCESS Statement . . . . .	E-5
Table E-3	BBPARM Member AAOALS00 - MAINVIEW AutoOPERATOR Only . . . . .	E-6
Table E-4	BBPARM Member AAOALT00 - MAINVIEW AutoOPERATOR Only . . . . .	E-9
Table E-5	BBPARM Member AAOARP00 - MAINVIEW AutoOPERATOR Only . . . . .	E-10
Table E-6	BBPARM Member AAOEWD00 - MAINVIEW AutoOPERATOR Only . . . . .	E-12
Table E-7	BBPARM Member AAOEXP00 - MAINVIEW AutoOPERATOR Only . . . . .	E-13
Table E-8	BBPARM Member AAOGME00 - MAINVIEW AutoOPERATOR Only . . . . .	E-17
Table E-9	BBPARM Member AAOMQL00 - MAINVIEW AutoOPERATOR Only . . . . .	E-20
Table E-10	BBPARM Member AAONCD00 - MAINVIEW AutoOPERATOR Only . . . . .	E-23
Table E-11	BBPARM Member AAOPRM00 - MAINVIEW AutoOPERATOR Only . . . . .	E-24

---

Table E-12	BBPARM Member AAOTRN00 - MAINVIEW AutoOPERATOR Only .....	E-29
Table E-13	BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only .....	E-31
Table E-14	BBPARM Member BBIVAR00 - MAINVIEW AutoOPERATOR Only .....	E-38
Table F-1	BBSAMP Data Set Members for MAINVIEW AutoOPERATOR .....	F-1
Table F-2	BBSAMP Data Set Members for MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS .....	F-3
Table F-3	BBSAMP Data Set Members for MAINVIEW AutoOPERATOR for IMS, MAINVIEW for IMS, and MAINVIEW for DBCTL .....	F-4
Table I-1	BBI-2 Product Library Configuration .....	I-9
Table K-1	BBI Control Commands Available to MAINVIEW AutoOPERATOR to PATROL EM Interface for VTAM Connections .....	K-4
Table K-2	BBPARM Member AAOALT00 .....	K-5
Table K-3	Output Messages Description .....	K-7

---

---

# About This Book

This book contains procedures for customizing the MAINVIEW AutoOPERATOR product to meet the needs of your site. The MAINVIEW AutoOPERATOR product is integrated with BBI architecture. The BBI architecture is a base architecture that enables authorized users to interrogate any OS/390, CICS, IMS, or DB2 subsystem in a VTAM network from a single terminal.

To install and customize MAINVIEW AutoOPERATOR, follow the instructions in the following books:

- *OS/390 and z/OS Installer Guide* to load the BMC Software product libraries
- *MAINVIEW Common Customization Guide* and the *MAINVIEW Administration Guide* to tailor BBI-based functions
- *Using MAINVIEW* guide to learn how to use MAINVIEW and become familiar with the MAINVIEW architecture
- *MAINVIEW AutoOPERATOR Customization Guide* to customize MAINVIEW AutoOPERATOR for your site's requirements

**Note:** You *must* customize the BBI-SS PAS before you customize MAINVIEW AutoOPERATOR.

This book is intended for the system programmer who needs to know how to modify the basic BBI product installation to include MAINVIEW AutoOPERATOR functions.

Throughout this book, references to OS/390 support also include support for MVS and z/OS.

---

# How This Book Is Organized

This book is organized as follows. In addition, this book contains an index.

<b>Chapter/Appendix</b>	<b>Description</b>
Chapter 1, "Migrating from MAINVIEW AutoOPERATOR Version 6.3.00 to Version 6.3.01"	describes how to migrate from MAINVIEW AutoOPERATOR 6.3 to MAINVIEW AutoOPERATOR 6.3.01
Chapter 2, "Migrating from MAINVIEW AutoOPERATOR Version 6.2.00 to Version 6.3.01"	describes how to migrate from MAINVIEW AutoOPERATOR 6.2 to MAINVIEW AutoOPERATOR 6.3
Chapter 3, "Migrating from MAINVIEW AutoOPERATOR Version 6.1.00 to Version 6.3.01"	describes how to migrate from MAINVIEW AutoOPERATOR 6.1 to MAINVIEW AutoOPERATOR 6.3
Chapter 4, "Continuous Operation: Using the Dynamic Parameter Manager"	describes how to use the Dynamic Parameter Manager for continuous operation
Chapter 5, "MVS Console Considerations"	describes how to allocate MVS consoles to MAINVIEW AutoOPERATOR subsystems that are used by TS users and EXECs
Chapter 6, "Implementing TapeSHARE"	describes how to implement the MAINVIEW AutoOPERATOR TapeSHARE option
Chapter 7, "Implementing Automatic Restart Manager (ARM)"	describes implementation considerations for the Automatic Restart Manager (ARM) and how to implement support for it
Chapter 8, "Implementing Support for MPF"	describes how MAINVIEW AutoOPERATOR can coexist with the MVS Message Processing Facility (MPF)
Chapter 9, "Activating the REXX/370 Alternate Library"	describes how to implement the REXX/370 Alternate Library
Chapter 10, "Defining OSPI Virtual Terminals"	describes how to define OSPI virtual terminals
Chapter 11, "Implementing BBI-SS PAS to CICS Communication"	describes how to implement BBI-SS PAS to CICS communication
Chapter 12, "MAINVIEW AutoOPERATOR for CICS Optional Tailoring"	describes optional steps for tailoring MAINVIEW AutoOPERATOR for CICS to meet the requirements of your site
Chapter 13, "Implementing MAINVIEW Products in IMS"	describes how to install and implement BMC Software modules, which are required for MAINVIEW AutoOPERATOR for IMS and MAINVIEW for IMS (MVIMS) Online
Chapter 14, "Implementing MAINVIEW AutoOPERATOR Access NV"	describes how to install the MAINVIEW AutoOPERATOR Access NV option into NetView and how to define the default NetView system to MAINVIEW AutoOPERATOR
Chapter 15, "Implementing AutoRULE"	describes how to implement the optional, advanced customization step, AutoRULE

<b>Chapter/Appendix</b>	<b>Description</b>
Chapter 16, "Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM"	describes how to customize MAINVIEW AutoOPERATOR to send ALERTs to the PATROL Enterprise Manager (PATROL EM) outboard processor
Chapter 17, "Implementing MAINVIEW AutoOPERATOR Advanced JES3 Support"	describes how to implement full JES3 support
Chapter 18, "Implementing Continuous State Manager"	describes how to implement Continuous State Manager (CSM) and how to attain the best possible performance
Chapter 19, "Specific MAINVIEW AutoOPERATOR Parameters and JCL"	describes JCL and parameters that are use to implement MAINVIEW AutoOPERATOR functions
Chapter 20, "Accessing Advanced SYSPROG Services from MAINVIEW AutoOPERATOR"	describes how to access the Advanced SYSPROG services from MAINVIEW AutoOPERATOR
Chapter 21, "Authorization for Special MAINVIEW AutoOPERATOR Functions"	describes the authorization that is required to issue commands or to use applications against a BBI-SS PAS target system
Chapter 22, "Authorizing EXECs Scheduled by an Operator or the Rule Processor"	describes how to authorize an EXEC that is scheduled by an operator or the Rule processor
Chapter 23, "Implementing MAINVIEW AutoOPERATOR for CICS Function Security (CMRSECU)"	discusses Function Security for MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS
Appendix A, "Issuing Commands from Terminal Sessions"	describes how to use MAINVIEW AutoOPERATOR to issue OS/390, IMS, CICS, and DBCTL commands from a terminal session (TS) command line
Appendix B, "Changing the Use of CICS Recoverable Temporary Storage"	describes how to change the use of CICS recoverable temporary storage
Appendix C, "CICS Transaction and Program Names"	describes the transaction and program names generated by MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS
Appendix D, "Manual Activation of BBI-SS to CICS Communication"	describes how to manually implement BBI-SS to CICS communication
Appendix E, "BBPARM Data Set Members for MAINVIEW AutoOPERATOR"	describes BBPARM data set members that you can use as examples to set up and customize MAINVIEW AutoOPERATOR
Appendix F, "Sample Data Set Members"	contains sample data set members for BBSAMP
Appendix G, "Expanding the PROFILE Pool Data Set"	discusses how to use the XTLOAD utility that is located in BBSAMP member XTJCL
Appendix H, "AutoRULE Worksheet"	contains a five-part worksheet that needs to be completed before you run the AutoCustomization step

Chapter/Appendix	Description
Appendix I, "How Product Libraries Should Be Used"	provides information about which product libraries you use when you make changes to a product and explains how the product libraries are created, and what their intended use is
Appendix J, "OSPI Terminal Definitions for CICS and IMS"	provides information about defining OSPI virtual terminals to CICS and IMS
Appendix K, "Establishing a VTAM Connection for MAINVIEW AutoOPERATOR to PATROL EM Communication"	describes how to establish a VTAM connection to enable MAINVIEW AutoOPERATOR to PATROL EM communication

## MAINVIEW AutoOPERATOR Product Library

MAINVIEW AutoOPERATOR is available with the following options:

- MAINVIEW AutoOPERATOR for OS/390
- MAINVIEW AutoOPERATOR for IMS
- MAINVIEW AutoOPERATOR for CICS
- MAINVIEW AutoOPERATOR Access NV
- MAINVIEW AutoOPERATOR TapeSHARE
- MAINVIEW AutoOPERATOR for MQSeries
- MAINVIEW AutoOPERATOR for SAP High Availability
- MAINVIEW AutoOPERATOR Elan Workstation

The base product and these options are documented in the following MAINVIEW AutoOPERATOR books:

- *MAINVIEW AutoOPERATOR Customization Guide*
- *MAINVIEW AutoOPERATOR Basic Automation Guide*
- *MAINVIEW AutoOPERATOR Advanced Automation Guide*
- *MAINVIEW AutoOPERATOR Options User Guide*
- *MAINVIEW AutoOPERATOR for MQSeries Installation and User Guide*
- *MAINVIEW AutoOPERATOR Solutions Guide*
- *MAINVIEW AutoOPERATOR Reference Summary*

---

This book also makes several references to the BBI architecture, which provides subsystem communication in its own OS/390 address space. The BBI online environment is described in the following books:

- *MAINVIEW Common Customization Guide*
- *MAINVIEW Administration Guide*
- *Using MAINVIEW*

## Recommended Documentation

Before using this customization guide, you must install the product as described in the *OS/390 and z/OS Installer Guide*. BBI-based functions can then be modified as described in the *MAINVIEW Common Customization Guide* and the *MAINVIEW Administration Guide*. You might also review the *Using MAINVIEW* book to become familiar with the MAINVIEW architecture.

## Related Documentation

Additionally, you might refer to the following IBM documentation:

- *Advanced Communications Function for VTAM (ACF/VTAM)*, SC38-0256
- *IBM System Modification Program Extended*, SC28-1107
- *ISPF and ISPF/PDF Version 2 for MVS Installation and Customization Guide*, SC34-4117
- *Advanced Communications Function for VTAM - Installation and Resource Definition*, SC27-0610

---

## Product Library

The BBI-based products include

- MAINVIEW AutoOPERATOR
- MAINVIEW for CICS
- MAINVIEW for IMS
- MAINVIEW FOCAL POINT
- MAINVIEW for DB2

Instructions for installing the product libraries for these products are in the *OS/390 and z/OS Installer Guide*. Installation and administration instructions for the BBI-based functions are in the *MAINVIEW Common Customization Guide* and *MAINVIEW Administration Guide*. Product-specific customization instructions are in the following books:

- *MAINVIEW AutoOPERATOR Customization Guide*
- *MAINVIEW for CICS Customization Guide*
- *MAINVIEW for IMS Customization Guide*
- *MAINVIEW FOCAL POINT User Guide*
- *MAINVIEW for DB2 Customization Guide*

How to use BBI General Services is documented in the following books:

- *MAINVIEW AutoOPERATOR Basic Automation Guide*
- *MAINVIEW for CICS PERFORMANCE REPORTER User Guide*
- *MAINVIEW for DB2 User Guide*
- *MAINVIEW for IMS Online - Analyzers Reference Manual*
- *MAINVIEW for IMS Online - Monitors and Traces Reference Manual*
- *Using MAINVIEW*

## What the Conventions Are

The following syntax notation is used in this book. Do not enter the special characters.

- Brackets, [ ], enclose optional parameters or keywords.
- Braces, { }, enclose a list of parameters; one must be chosen.
- A vertical line, |, separates alternative options; one can be chosen.
- An *italicized* or underlined parameter is the default.
- AN ITEM IN CAPITAL LETTERS must be entered exactly as shown.
- Items in lowercase letters are values that you supply.

---

---

# Chapter 1 Migrating from MAINVIEW AutoOPERATOR Version 6.3.00 to Version 6.3.01

This chapter contains the following discussions:

- information that you need to know to migrate quickly and successfully from MAINVIEW AutoOPERATOR 6.3.00 to MAINVIEW AutoOPERATOR 6.3.01
- general questions and answers about the current release
- installation considerations for MAINVIEW AutoOPERATOR 6.3.01

You should read this chapter in conjunction with the MAINVIEW AutoOPERATOR version 6.3.01 Release Notes.

## What You Need to Know

The following sections include important discussions about

- what new product options have been added
- what changes have been made to existing parameters, Rules, and EXECs
- SAP High Availability issues
- Total Object Manager issues
- MAINVIEW AutoOPERATOR 6.3.01 overhead reduction

## New Product Option

This release includes a new product option, Total Object Manager. This option provides a basic infrastructure through MAINVIEW AutoOPERATOR applications that you can use to manage objects within a sysplex or across sysplexes. MAINVIEW AutoOPERATOR 6.3.01 release includes the first application, Start Task Manager (STM), that you can use to manage the start, stop, and other statuses of Started Tasks.

For more information about this product option, refer to the *MAINVIEW Total Object Manager User Guide*.

## Changes Made to Existing Parameters, Rules, and EXECs

All existing MAINVIEW AutoOPERATOR 6.3.00 parameter values are honored in this release. You can copy your parameter members from your MAINVIEW AutoOPERATOR 6.3.00 BBI-SS PAS and use them for your MAINVIEW AutoOPERATOR 6.3.01 PAS.

You must activate any new parameters or parameter values either by editing the BBPARAM member, changing the value, and restarting the BBI-SS PAS, or by using the Dynamic Parameter Manager application (which does not require a BBI-SS PAS restart).

In addition, you can run your automation without any manual modifications by copying your existing Rules and EXECs to your new MAINVIEW AutoOPERATOR 6.3.01 PAS.

## SAP High Availability Issues

If you are using the MAINVIEW AutoOPERATOR SAP High Availability product, you need to perform Step 6A on page 32-12 of the *MAINVIEW AutoOPERATOR Options User Guide*.

## Total Object Manager Issues

BMC Software strongly encourages customers to use the MAINVIEW Total Object Manager (TOM) application for managing their Started Tasks. To use the MAINVIEW Total Object Manager, you must

- Apply IBM PTF UA00713 to your system, which helps you avoid a scrolling problem on the TOM Started Task Definition panel.
- Apply IBM APAR OA03333, which helps you avoid getting an abend when using the online help in TOM.
- Allocate a pair of log data sets and a registry data set, tailor the data sets appropriately, and submit the TOMALLOC JCL member that was supplied in the BBSAMP library.

The user interface for TOM is written in compiled REXX. Not all environments can execute compiled REXX procedures. If your site cannot handle compiled REXX, you need to prepare your environment by following the procedures found in Chapter 9, “Activating the REXX/370 Alternate Library.”

For additional customization information, refer to Chapter 2 of the *MAINVIEW Total Object Manager User Guide*.

**Note:** The MAINVIEW AutoOPERATOR Continuous State Manager (CSM) documentation has been moved to the *MAINVIEW AutoOPERATOR Solutions Guide*. Support for this application will be dropped in a future release of MAINVIEW AutoOPERATOR.

# General Questions and Answers about the New Release

The following sections contain information about MAINVIEW AutoOPERATOR 6.3.01.

## What Has Changed about Rules

No changes were made to Rules in MAINVIEW AutoOPERATOR 6.3.01.

## What DD Names Have Changed

No changes were made to existing DD names in MAINVIEW AutoOPERATOR 6.3.01.

## What Parameters Have Changed

No changes were made to parameters in MAINVIEW AutoOPERATOR 6.3.01.

## How to Secure MAINVIEW AutoOPERATOR 6.3.01

Just as with MAINVIEW AutoOPERATOR 6.3.00, you can implement security by using the External Security Manager (ESM). While you can continue to use BMC Software proprietary security (documented in *Implementing Security for MAINVIEW Products*), you might want to control security of MAINVIEW AutoOPERATOR through your ESM.

This release includes no additional security options.

## How Much CPU Does the New Release Use

The new release does not use a significantly higher amount of CPU than the previous release.

## **Does MAINVIEW AutoOPERATOR 6.3.01 Use Common Storage Differently**

MAINVIEW AutoOPERATOR 6.3.01 does not use common storage differently from MAINVIEW AutoOPERATOR 6.3.00.

## **Does MAINVIEW AutoOPERATOR 6.3.01 Use Private Storage Differently**

MAINVIEW AutoOPERATOR 6.3.01 does not use private storage differently from MAINVIEW AutoOPERATOR 6.3.00.

## **Does This Version Communicate with Version 6.3.00**

Any BBI-SS PAS with an activated MAINVIEW AutoOPERATOR 6.3.01 product can schedule an ALERT or EXEC to any MAINVIEW AutoOPERATOR 6.2.00 BBI-SS PAS. In addition, a MAINVIEW AutoOPERATOR 6.3.01 terminal session can access MAINVIEW AutoOPERATOR 6.2.00 panels.

You can install MAINVIEW AutoOPERATOR 6.3.01 on your test system and use it to review production operation until you are satisfied with the product's stability and functionality. Then you can migrate MAINVIEW AutoOPERATOR 6.3.01 through your other systems, one at a time, at a pace that is acceptable to your staff.

## **Does Version 6.2 Communicate with This Version**

ALERTs, EXECs, and Rules can be sent from a MAINVIEW AutoOPERATOR 6.2 BBI-SS PAS to a MAINVIEW AutoOPERATOR 6.3.01 BBI-SS PAS.

## **Which IBM Software Does MAINVIEW AutoOPERATOR 6.3.01 Require**

Refer to Chapter 1, "Installation Prerequisites," in the *MAINVIEW Installation Requirements Guide* for information about the IBM software (and levels) that is required to run MAINVIEW AutoOPERATOR 6.3.01.

## What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01

The following BMC Software products are supported by MAINVIEW AutoOPERATOR 6.3.01 at these version levels *or later*:

- SYSPROG Services version 3.2.01
- MAINVIEW for CICS version 5.4.00
- MAINVIEW for IMS version 3.3.20
- MAINVIEW for DBCTL version 3.3.20
- MAINVIEW for OS/390 version 2.6.00

## What Is Not Supported by Version 6.3.01

MAINVIEW AutoOPERATOR 6.3.01 does not support any IBM software release that is not supported by IBM, which includes

- IMS/ESA release earlier than version 7.0
- CICS Transaction Server release earlier than version 1.3
- MQSeries release earlier than version 1.2
- NetView versions 1, 2, and 3
- OS/390 releases earlier than version 2.10
- Tivoli NetView for z/OS releases earlier than version 5.1

MAINVIEW AutoOPERATOR 6.3.01 does not support

- COR (CommandMQ On Ramp)
- STRATEGY=QUALIFIED Rules

## What Functions Will Not Be Supported in the Next Release of MAINVIEW AutoOPERATOR

Support for communicating with PATROL Enterprise Manager using a VTAM printer (see BBPARM member AAOALTxx) will be removed in the next release. This communication method is used to send AutoOPERATOR Alerts to PATROL EM. An I/O Concepts X-Direct coax gateway is required for PATROL EM to emulate a VTAM printer.

The more robust TCP/IP-based communication interface with PATROL EM is the recommended method (see Chapter 16, “Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM”) for synchronizing AutoOPERATOR Alerts with PATROL EM. This method also removes the requirement for the I/O Concepts hardware.

## What Functions Will Not Be Supported in Future Releases of MAINVIEW AutoOPERATOR

In a future release, BMC Software plans to discontinue the use of documented variables by the Continuous State Manager application.

When these documented variables are discontinued, BMC Software will attempt to identify all customer implementations that are dependent on these variables to provide an alternate method of functionality.

## Installation Considerations for MAINVIEW AutoOPERATOR 6.3.01

This section contains technique-specific installation considerations for migrating from previous releases of MAINVIEW AutoOPERATOR to MAINVIEW AutoOPERATOR version 6.3.01.

### Getting New BBKEYS

If a new release of MAINVIEW AutoOPERATOR is installed on the same CPU where a currently licensed copy resides, the new release uses the same set of BBKEYS as the current copy. You do *not* need to request a new set of BBKEYS. You are required to obtain a new set of BBKEYS only when changing or upgrading CPUs.

Refer to the *MAINVIEW Common Customization Guide* for more information about acquiring BBKEYS.

### Defining MAINVIEW AutoOPERATOR Resources to CICS

If you have customized your CICS regions for use with a previously released version of MAINVIEW AutoOPERATOR, the CICS resource definitions for your BMC Software products must be refreshed.

This update is accomplished by customizing and executing the BMC Software job CMRCSDES in the BBSAMP data set.

No other changes are needed except changing the CICS execution JCL to reflect the names of the new BBPARM data sets.

## Migrating by Using the OS/390 and z/OS Installer: Standard Installation

The *OS/390 and z/OS Installer Guide* refers to two installation methods:

- Standard installation provides a fast IEBCOPY installation process. This method offers an optional step that creates a predefined SMP/E environment for SMP/E maintenance. The Standard installation method requires less expertise than an SMP/E installation, but it offers less control over the SMP/E environment.
- SMP/E is an industry-standard installation program that provides a variety of capabilities for unloading products and maintenance. SMP/E provides more flexibility than the Standard installation method for setting up and using an SMP/E environment, but it is more complicated and time consuming.

If you are migrating by using standard installation with the BMC Software OS/390 and z/OS Installer, read the following section on implementing the JES2MAO exit again.

If you are replacing any MAINVIEW AutoOPERATOR release and all of the following statements are true, you must implement the JES2MAO exit again:

- Your site is licensed for the MAINVIEW AutoOPERATOR for OS/390 option.
- You elected to implement the JES2MAO exit.
- The JES2MAO exit is linked to the existing system's BBLINK library.

Refer to “Displaying MVS JES2 Job Numbers in Address Space Application” on page 19-4, for additional information about the JES2MAO exit.

## AutoCustomization Considerations When Replacing Existing Products

When replacing existing BMC Software products, the following conditions apply:

- All existing products that are replaced must be customized again.
- Any modifications or additions to the UBBPARM data set that were performed manually after running AutoCustomization on the existing system need to be performed as necessary on the new UBBPARM data set after it is created by AutoCustomization on the new system.

Refer to “Customize the Installed Products” in the *OS/390 and z/OS Installer Guide* for additional information about AutoCustomization.

## Migrating by Using System Maintenance Program (SMP)

Read this section if you are using the System Maintenance Program (SMP) to migrate.

If MAINVIEW AutoOPERATOR 6.3.01 is installed into a set of SMP zones that contains any release of the BMC Software products listed in “What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01” on page 1-6, the minimum release level of those products must also be installed.



---

---

# Chapter 2 Migrating from MAINVIEW AutoOPERATOR Version 6.2.00 to Version 6.3.01

This chapter contains the following discussions:

- information that you need to know to migrate quickly and successfully from MAINVIEW AutoOPERATOR 6.2.00.00 to MAINVIEW AutoOPERATOR 6.3.01
- general questions and answers about the current release
- installation considerations for MAINVIEW AutoOPERATOR 6.3.01

You should read this chapter in conjunction with the MAINVIEW AutoOPERATOR version 6.3.01 Release Notes.

## What You Need to Know

The following sections include important discussions about

- what new product options have been added
- what changes have been made to existing parameters, Rules, and EXECs
- CSM compatibility issues
- MAINVIEW AutoOPERATOR 6.3.01 overhead reduction
- MAINVIEW AutoOPERATOR and PATROL Enterprise Manager (PATROL EM) connections

## New Product Option

This release includes a new product option, MAINVIEW AutoOPERATOR for SAP High Availability.

This product option provides continuous availability for critical SAP resources. For more information about this option, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

## Changes Made to Existing Parameters, Rules, and EXECs

Support of MVS System Symbols in BBPARM members has been added. This support enables you to reduce the number of PARMLIB members that are required to run MAINVIEW AutoOPERATOR products. For more information about substituting system symbols for system variables, refer to “System Variables Substitution in Parameter Library Members” on page E-4.

In addition, a control card (PROCESS statement) has been added so that you can control variable substitution on a member-by-member basis. For more information about the PROCESS statement, refer to “System Variables Substitution in Parameter Library Members” on page E-4.

Otherwise, all existing MAINVIEW AutoOPERATOR 6.2.00 parameter values are honored in this release. You can copy your parameter members from your MAINVIEW AutoOPERATOR 6.2.00 BBI-SS PAS and use them for your MAINVIEW AutoOPERATOR 6.3.01 PAS.

You must activate any new parameters or parameter values either by editing the BBPARM member, changing the value, and restarting the BBI-SS PAS, or by using the Dynamic Parameter Manager application (which does not require a BBI-SS PAS restart).

In addition, you can run your automation without any manual modifications by copying your existing Rules and EXECs to your new MAINVIEW AutoOPERATOR 6.3.01 PAS.

Some compatibility limitations exist between MAINVIEW AutoOPERATOR 6.2.00 Rules and 6.3.01 Rules. Refer to “What Has Changed about Rules” on page 2-4.

For additional information about new parameters or new BBPARM members, refer to “What Parameters Have Changed” on page 2-5.

## CSM Compatibility Issues

No CSM compatibility issues occur when you migrate from MAINVIEW AutoOPERATOR 6.2.00 to 6.3.01.

## MAINVIEW AutoOPERATOR 6.3.01 Overhead Reduction

By eliminating the use of Process Queue Elements (PQEs) for minor messages (real and simulated minor messages), the overhead incurred by MAINVIEW AutoOPERATOR is reduced. In addition, outages caused by MAXPQE exhaustion are reduced.

## MAINVIEW AutoOPERATOR and PATROL Enterprise Manager Connections

Support has been added to enable you to manage MAINVIEW AutoOPERATOR/PATROL Enterprise Manager (PATROL EM) connections from the PATROL EM AO Connection side. This enhancement helps to ensure higher availability of your automation and ALERT systems. This support is available in PATROL EM AO Connection version 2.0.00.

For more information, refer to the *PATROL Enterprise Manager AO Connection User Guide*.

# General Questions and Answers about the New Release

The following sections contain information about MAINVIEW AutoOPERATOR 6.3.01.

## What Has Changed about Rules

By using Rules instead of EXECs for certain automation tasks, you can reduce CPU usage by as much as 90 percent.

**Note:** The Rules and Rule Sets that you created on a MAINVIEW AutoOPERATOR 6.2.00 PAS can run without changes on MAINVIEW AutoOPERATOR 6.3.01.

With MAINVIEW AutoOPERATOR 6.3.01, the following changes apply to your MAINVIEW AutoOPERATOR 6.2.00 Rules and Rule Sets:

- A terminal session connected to a MAINVIEW AutoOPERATOR 6.2.00 PAS can target a MAINVIEW AutoOPERATOR 6.3.01 PAS (and conversely). It can enable, disable, browse, edit, and otherwise manipulate MAINVIEW AutoOPERATOR 6.3.01 Rules.
- When viewing a 6.2.00 Rule from a MAINVIEW AutoOPERATOR 6.3.01 PAS, all Help panels contain information specific to what is available to a 6.3.01 Rule.
- On the Action Specification panel, the **EXEC Name/Parms** field has been expanded from one line to two lines, enabling you to pass more parameters to an EXEC. Although Rules can be created, updated and deleted by using any combination of MAINVIEW AutoOPERATOR versions, you should be cautious when you use an earlier version of MAINVIEW AutoOPERATOR to update or edit a Rule that was created in MAINVIEW AutoOPERATOR version 6.3 or later. The results can be unpredictable because only one line of the **EXEC Name/Parms** field can be displayed in earlier versions.
- Rule ID PM0010I has been added to Rule set AAORULM1. To enable the MVS solution, you must enable this Rule and Rule Set.

- Rules ALRMSTRT and ALRMSTOP in Rule Set AAORUL00 were shipped *enabled* in earlier versions of MAINVIEW AutoOPERATOR. The default for these Rules has been changed to *disabled*. If you want to implement the Alarm Solution that is documented in the *MAINVIEW AutoOPERATOR Solution Guide* “Activating the MVALARM Solution,” you must enable these Rules.

**Note:** If you are currently using the distributed Rules ALRMSTRT and ALRMSTOP to create (or delete) a MAINVIEW AutoOPERATOR ALERT, you should review the automation that uses these Rules, especially if you have also specified ALRTRCVE=YES in BBPARM member BBISSP00. If YES is specified, the PAS receives product ALERTs from other BMC Software products, which might cause duplicate ALERTs to be created. Refer to the *MAINVIEW Common Customization Guide* for more information about parameters set in BBPARM member BBISSP00.

## What Are the Changes Made to Variables

A shared variable (*RREPJES*) has been added to BBPARM member BBIVARxx. You can use this variable to supply the JES version when you execute the RREPRX EXEC before JES is initialized. For more information, refer to “Using the MAINVIEW AutoOPERATOR-Supplied Utility EXECs” in the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

## What DD Names Have Changed

No changes were made to existing DD names in MAINVIEW AutoOPERATOR 6.3.01.

## What Parameters Have Changed

The following changes were made to BBPARM members in MAINVIEW AutoOPERATOR 6.3.01:

In BBPARM member AAOEXPxx, support for the specification REXX=NO has been discontinued. Therefore, the REXX support option has been removed from the Dynamic Parameter Manager (EXEC Parameters) panel. For more information about Dynamic Parameter Manager, refer to Chapter 4, “Continuous Operation: Using the Dynamic Parameter Manager.”

## What Are the Changes Related to IMS

The following region types have been added to the IMS Region Operator Console Display (OCD):

**JBP**                    non-message-driven IMS JAVA applications region

**JMP**                    message-driven IMS JAVA applications region

These region types support the new Java address spaces in IMS 8.1. For more information, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

## What Are the Changes Related to CICS

A new IMFEXEC command, IMFEXEC CICS ALTER DUMPCODE, has been added, which specifies whether to secure the alteration of CICS SYSTEM and TRANSACTION DUMPCODES. Refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for more information about the syntax of this command.

## What Else Has Changed

In addition, MAINVIEW AutoOPERATOR version 6.3.01 contains the following changes:

- Additional IMFCC return codes for the IMFEXEC ALERT FUNCTION keyword have been added. For more information, refer to “Return Codes for FUNCTION Keywords” in the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.
- QAODUMP, a new distributed utility EXEC, has been added. For more information, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

## How to Secure MAINVIEW AutoOPERATOR 6.3.01

Just as with MAINVIEW AutoOPERATOR 6.2.00, you can implement security by using the External Security Manager (ESM). While you can continue to use BMC Software proprietary security (documented in *Implementing Security for MAINVIEW Products*), you might want to control security of MAINVIEW AutoOPERATOR through your ESM.

This release includes no additional security options.

## How Much CPU Does the New Release Use

The new release does not use a significantly higher amount of CPU than the previous release.

## Does MAINVIEW AutoOPERATOR 6.3.01 Use Common Storage Differently

MAINVIEW AutoOPERATOR 6.3.01 does not use common storage differently from MAINVIEW AutoOPERATOR 6.2.00.

## Does MAINVIEW AutoOPERATOR 6.3.01 Use Private Storage Differently

MAINVIEW AutoOPERATOR 6.3.01 does not use private storage differently from MAINVIEW AutoOPERATOR 6.2.00.

## Does This Version Communicate with Version 6.2.00

Any BBI-SS PAS with an activated MAINVIEW AutoOPERATOR 6.3.01 product can schedule an ALERT or EXEC to any MAINVIEW AutoOPERATOR 6.2.00 BBI-SS PAS. In addition, a MAINVIEW AutoOPERATOR 6.3.01 terminal session can access MAINVIEW AutoOPERATOR 6.2.00 panels.

You can install MAINVIEW AutoOPERATOR 6.3.01 on your test system and use it to review production operation until you are satisfied with the product's stability and functionality. Then you can migrate MAINVIEW AutoOPERATOR 6.3.01 through your other systems, one at a time, at a pace that is acceptable to your staff.

**Note:** Be sure to review compatibility restrictions in “What Has Changed about Rules” on page 2-4.

## Does Version 6.2.00 Communicate with This Version

ALERTs, EXECs, and Rules can be sent from a MAINVIEW AutoOPERATOR 6.2.00 BBI-SS PAS to a MAINVIEW AutoOPERATOR 6.3.01 BBI-SS PAS.

## Which IBM Software Does MAINVIEW AutoOPERATOR 6.3.01 Require

Refer to Chapter 1, “Installation Prerequisites,” in the *MAINVIEW Installation Requirements Guide* for information about the IBM software (and levels) that is required to run MAINVIEW AutoOPERATOR 6.3.01.

## What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01

The following BMC Software products are supported by MAINVIEW AutoOPERATOR 6.3.01 at these version levels or later:

- SYSPROG Services version 3.2.01
- MAINVIEW for CICS version 5.4.00
- MAINVIEW for IMS version 3.3.20
- MAINVIEW for DBCTL version 3.3.20
- MAINVIEW for OS/390 version 2.6.00

## What Is Not Supported by Version 6.3.01

MAINVIEW AutoOPERATOR 6.3.01 does not support any IBM software release that is not supported by IBM, which includes

- IMS/ESA release earlier than version 7.0
- CICS Transaction Server release earlier than version 1.3
- MQSeries release earlier than version 1.2
- NetView versions 1, 2, and 3
- OS/390 releases earlier than version 2.10
- Tivoli NetView for z/OS releases earlier than version 5.1

MAINVIEW AutoOPERATOR 6.3.01 does not support

- COR (CommandMQ On Ramp)
- STRATEGY=QUALIFIED Rules

## **What Functions Will Not Be Supported in the Next Release of MAINVIEW AutoOPERATOR**

Support for communicating with PATROL Enterprise Manager using a VTAM printer (see BBPARM member AAOALTxx) will be removed in the next release. This communication method is used to send AutoOPERATOR Alerts to PATROL EM. An I/O Concepts X-Direct coaxial gateway is required for PATROL EM to emulate a VTAM printer.

The more robust TCP/IP-based communication interface with PATROL EM is the recommended method (see Chapter 16, “Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM”) for synchronizing AutoOPERATOR Alerts with PATROL EM. This method also removes the requirement for the I/O Concepts hardware.

## **What Functions Will Not Be Supported in Future Releases of MAINVIEW AutoOPERATOR**

In a future release, BMC Software plans to discontinue the use of documented variables by the Continuous State Manager application.

When these documented variables are discontinued, BMC Software will attempt to identify all customer implementations that are dependent on these variables to provide an alternate method of functionality.

# Installation Considerations for MAINVIEW AutoOPERATOR 6.3.01

This section contains technique-specific installation considerations for migrating from previous releases of MAINVIEW AutoOPERATOR to MAINVIEW AutoOPERATOR version 6.3.01.

## Getting New BBKEYS

If a new release of MAINVIEW AutoOPERATOR is installed on the same CPU where a currently licensed copy resides, the new release uses the same set of BBKEYS as the current copy. You do *not* need to request a new set of BBKEYS. You are required to obtain a new set of BBKEYS only when changing or upgrading CPUs.

Refer to the *MAINVIEW Common Customization Guide* for more information about acquiring BBKEYS.

## Defining MAINVIEW AutoOPERATOR Resources to CICS

If you have customized your CICS regions for use with a previously released version of MAINVIEW AutoOPERATOR, the CICS resource definitions for your BMC Software products must be refreshed.

This update is accomplished by customizing and executing the BMC Software job CMRCSDES in the BBSAMP data set.

No other changes are needed except changing the CICS execution JCL to reflect the names of the new BBPARM data sets.

## Migrating by Using the OS/390 and z/OS Installer: Standard Installation

The *OS/390 and z/OS Installer Guide* refers to two installation methods:

- Standard installation provides a fast IEBCOPY installation process. This method offers an optional step that creates a predefined SMP/E environment for SMP/E maintenance. The Standard installation method requires less expertise than an SMP/E installation, but it offers less control over the SMP/E environment.

- SMP/E is an industry-standard installation program that provides a variety of capabilities for unloading products and maintenance. SMP/E provides more flexibility than the Standard installation method for setting up and using an SMP/E environment, but it is more complicated and time consuming.

If you are migrating by using standard installation with the BMC Software OS/390 and z/OS Installer, read the following section on implementing the JES2MAO exit again.

If you are replacing any MAINVIEW AutoOPERATOR release and all of the following statements are true, you must implement the JES2MAO exit again:

- Your site is licensed for the MAINVIEW AutoOPERATOR for OS/390 option.
- You elected to implement the JES2MAO exit.
- The JES2MAO exit is linked to the existing system's BBLINK library.

Refer to “Displaying MVS JES2 Job Numbers in Address Space Application” on page 19-4, for additional information about the JES2MAO exit.

## **AutoCustomization Considerations When Replacing Existing Products**

When replacing existing BMC Software products, the following conditions apply:

- All existing products that are replaced must be customized again.
- Any modifications or additions to the UBBPARM data set that were performed manually after running AutoCustomization on the existing system need to be performed as necessary on the new UBBPARM data set after it is created by AutoCustomization on the new system.

Refer to “Customize the Installed Products” in the *OS/390 and z/OS Installer Guide* for additional information about AutoCustomization.

## **Migrating by Using System Maintenance Program (SMP)**

Read this section if you are using the System Maintenance Program (SMP) to migrate.

If MAINVIEW AutoOPERATOR 6.3.01 is installed into a set of SMP zones that contains any release of the BMC Software products listed in “What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01” on page 2-8, the minimum release level of those products must also be installed.

---

---

# Chapter 3 Migrating from MAINVIEW AutoOPERATOR Version 6.1.00 to Version 6.3.01

This chapter contains the following discussions:

- information that you need to know to migrate quickly and successfully from MAINVIEW AutoOPERATOR 6.1.00 to MAINVIEW AutoOPERATOR 6.3.01
- general questions and answers about the current release
- installation considerations for MAINVIEW AutoOPERATOR 6.3.01

You should read this chapter in conjunction with the MAINVIEW AutoOPERATOR version 6.3.01 Release Notes.

## What You Need to Know

The following sections include important discussions about

- what new product options have been added
- what messages have changed
- what changes have been made to existing parameters, Rules, and EXECs
- CSM 6.1.00 to CSM 6.3.01 compatibility
- MAINVIEW AutoOPERATOR 6.3.01 overhead reduction
- MAINVIEW AutoOPERATOR and PATROL Enterprise Manager (PATROL EM) connections

## New Product Option

This release includes the following new product option: MAINVIEW AutoOPERATOR for SAP High Availability.

This product option provides continuous availability for critical SAP resources. For more information, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

## Messages That Have Changed

Messages that have changed since the MAINVIEW AutoOPERATOR 6.1.00 release are as follows:

- The following three TapeSHARE message IDs have changed:
  - DS0501I has been changed to DS4000I.
  - DS1201I has been changed to DS4200I.
  - DS1401I has been changed to DS4300I.

If you have any automation that is triggered from these messages, you must convert those Rules to use the new message IDs.

- New fields **Rule Set** and **ZZZZZZZZ**, which represent the Rule Set name, have been added to message DX9501I:

```
DX9501I  RULE: XXXXXXXX SUSPENDED AFTER YYY MATCHES; RULE
SET: ZZZZZZZZ
```

This message contains the new fields.

- Two new messages have been added:
  - AU6021E JES JOB NUMBER GREATER THAN 99,999

This message is issued the first time MAINVIEW AutoOPERATOR detects a job number greater than 99,999. For more information, refer to “IMFJNUM Option and Seven-Digit JES Job Numbers (z/OS v1.2 and Later)” on page 19-6.

- AU6020W 7 DIGIT JES JOB NUMBER DETECTED

This message appears the first time MAINVIEW AutoOPERATOR detects a seven-digit mode job number that is between 1 and 99,999. This WTO message warns you that the seven-digit job number is incompatible with MAINVIEW AutoOPERATOR.

---

## Changes Made to Existing Parameters, Rules, and EXECs

Support of MVS System Symbols in BBPARM members has been added. This support enables you to reduce the number of PARMLIB members that are required to run MAINVIEW AutoOPERATOR products. For more information about substituting system symbols for system variables, refer to “System Variables Substitution in Parameter Library Members” on page E-4.

In addition, a control card (PROCESS statement) has been added so that you can control variable substitution on a member-by-member basis. For more information about the Process statement, refer to “System Variables Substitution in Parameter Library Members” on page E-4.

Otherwise, all existing MAINVIEW AutoOPERATOR 6.1.00 parameter values are honored in this release. You can copy your parameter members from your MAINVIEW AutoOPERATOR 6.1.00 BBI-SS PAS and use them for your MAINVIEW AutoOPERATOR 6.3.01 PAS.

You must activate any new parameters or parameter values either by editing the BBPARM member, changing the value, and restarting the BBI-SS PAS, or by using the Dynamic Parameter Manager application (which does not require a BBI-SS PAS restart).

In addition, you can run your automation without any manual modifications by copying your existing Rules and EXECs to your new MAINVIEW AutoOPERATOR 6.3.01 PAS.

Some compatibility limitations exist between MAINVIEW AutoOPERATOR 6.1.00 Rules and 6.3.01 Rules. Refer to “What Has Changed about Rules” on page 3-5.

For additional information about new parameters or new BBPARM members, refer to “What Parameters Have Been Changed or Added” on page 3-8.

## CSM Compatibility Issues

No CSM compatibility issues exist when you migrate from MAINVIEW AutoOPERATOR 6.1.00 to 6.3.01, but a new CSM primary command is available. You can use the PROFile command on the Global Overview panel to select the names of the groups that are to be displayed in the Global Overview. For more information, refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

In addition, objects can be moved from an inactive to an active group within the same repository. This feature enables you to recover Started Task objects from failed LPARs in the SYSPLEX. From an inactive group, Moves can be triggered either from the Global Overview panel, with the CSM command line interface or by scheduling CSMACT on the BBI-PAS to where you want to move the object. For more information, refer to “Chapter 11, Managing Critical Applications with Continuous State Manager” in the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

## **MAINVIEW AutoOPERATOR 6.3.01 Overhead Reduction**

By eliminating the use of Process Queue Elements (PQEs) for minor messages (real and simulated minor messages), the overhead incurred by MAINVIEW AutoOPERATOR is reduced. In addition, outages caused by MAXPQE exhaustion are reduced.

## **MAINVIEW AutoOPERATOR and PATROL Enterprise Manager Connections**

Support has been added to enable you to manage MAINVIEW AutoOPERATOR/PATROL Enterprise Manager (PATROL EM) connections from the PATROL EM AO Connection side. This enhancement helps to ensure higher availability of your automation and ALERT systems. This support is available in PATROL EM AO Connection version 2.0.

For more information, refer to the *PATROL Enterprise Manager AO Connection User Guide*.

# General Questions and Answers about the New Release

The following sections contain discussions about migrating from MAINVIEW AutoOPERATOR 6.1.00 to MAINVIEW AutoOPERATOR 6.3.01.

## What Has Changed about Rules

For certain automation tasks, using Rules instead of EXECs has proven to reduce CPU usage by as much as 90 percent.

**Note:** The Rules and Rule Sets that you created on a MAINVIEW AutoOPERATOR 6.1.00 PAS can run without changes on MAINVIEW AutoOPERATOR 6.3.01.

With MAINVIEW AutoOPERATOR 6.3.01, the following changes apply to your MAINVIEW AutoOPERATOR 6.1.00 Rules and Rule Sets:

- A terminal session connected to a MAINVIEW AutoOPERATOR 6.1.00 PAS can target an MAINVIEW AutoOPERATOR 6.3.01 PAS (and conversely). It can enable, disable, browse, edit, and otherwise manipulate MAINVIEW AutoOPERATOR 6.3.01 Rules.
- When viewing a 6.1.00 Rule from a MAINVIEW AutoOPERATOR 6.3.01 PAS, all Help panels contain information specific to what is available for a 6.3.01 Rule.

- The 6.3.01 Rule-creation panels can be used to add new version 6.1.00 Rules with the following conditions:

The MAINVIEW AutoOPERATOR 6.3.01 Rule-creation panels contain fields supported in MAINVIEW AutoOPERATOR version 6.3.01. Use of some of these fields when creating a version 6.1.00 Rule will return error messages when the Rule is created. When the other fields are used, although no error message is displayed, the Rule might not fire, or might fire incorrectly, such as when using the new **REWORD CMD** field for CMD Rules.

- On the Action Specification panel, the **EXEC Name Parameters** field has been expanded from one line to two lines, enabling you to pass more parameters to an EXEC. Although Rules can be created, updated and deleted by using any combination of MAINVIEW AutoOPERATOR versions, you should be cautious when you use an earlier version of MAINVIEW AutoOPERATOR to update or edit a Rule that was created in MAINVIEW AutoOPERATOR 6.3 or later. The results can be unpredictable because only one line of the **EXEC Name/Parms** field can be displayed in earlier versions.
- Rule ID PM0010I has been added to Rule set AAORULM1. To enable the MVS solution, you must enable this Rule and Rule Set.
- Rules ALRMSTRT and ALRMSTOP in Rule Set AAORUL00 were shipped *enabled* in earlier versions of MAINVIEW AutoOPERATOR. The default for these Rules has been changed to *disabled*. If you want to implement the Alarm Solution that is documented by BMC Software, you must enable these Rules.

**Note:** If you are currently using the distributed Rules ALRMSTRT and ALRMSTOP to create (or delete) a MAINVIEW AutoOPERATOR ALERT, you should review the automation that uses these Rules, especially if you have also specified ALRTRCVE=YES in BBPARM member BBISSP00. If YES is specified, the PAS receives ALERTs from other BMC Software products, which might cause duplicate ALERTs to be created. Refer to the *MAINVIEW Common Customization Guide* for more information about parameters set in BBPARM member BBISSP00.

Other changes made to Rules in MAINVIEW AutoOPERATOR are as follows:

- A new field, **REWORD CMD**, has been added to the Action Specification Panel for CMD events. With this field, you can enter a new command that replaces the original command and the response returns to the originating console. Prior to this release, you were restricted to rejecting the command and reissuing the command CONSOLE 00.

- The Filter Criteria panel has been modified so that you can specify a match rate for a Rule Set. When Rules within a specified Rule Set exceed the match rate, the Rule can be suspended or disabled, or the Rule will fire, but the actions specified by the Rule are not taken until the rate falls below the threshold. This modification allows you to specify a match rate for all Rules within the Rule Set that do not have individual match rates specified.
- The **Ignore leading plus** field has been added to the selection criteria for MSG types. This feature enables messages that are issued from authorized and nonauthorized programs to be trapped by a single Rule.

## What Are the Changes Made to Variables

The following changes were made to variables in MAINVIEW AutoOPERATOR 6.3.01.

A shared variable (*RREPJES*) has been added to BBPARM member BBIVARxx. You can use this variable to supply the JES version when you execute the RREPRX EXEC before JES is initialized. For more information, refer to “Using the AutoOPERATOR-Supplied Utility EXECs” in the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

Other changes are as follows:

- New variables are returned when you use the IMFEXEC WAITLIST command. The variables return the name, date, and time of the waiting EXEC.
- New ALRT.x. variables are returned with the AOEXEC SYSINFO command. These variables might contain either the value YES or the value NO, indicating whether a particular subsystem has ALRTRCVE=YES specified in its BBISSPxx member.
- Another new variable returned by AOEXEC SYSINFO is *IMFXCFGP*. When returned, it contains the value of the XCF group of the addressed systems.

For more information about these variables, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

## What DD Names Have Changed

No changes were made to existing DD names in MAINVIEW AutoOPERATOR 6.3.01.

## What Parameters Have Been Changed or Added

The following changes were made to BBPARM members in MAINVIEW AutoOPERATOR 6.3.01:

In BBPARM member AAOEXPxx, support for the specification REXX=NO has been discontinued. Therefore, the REXX support option has been removed from the Dynamic Parameter Manager (EXEC Parameters) panel. For more information about Dynamic Parameter Manager, refer to “Continuous Operation: Using the Dynamic Parameter Manager” on page 4-1.

Other changes are as follows:

- BBPARM member BBISSP00 can now have a user-defined suffix, which can be specified during BBI-SS PAS initialization. The default suffix is 00. Specifying different member name suffixes enables multiple BBI-SS PASs to process different configuration members from the same BBPARM library. Refer to the *MAINVIEW Common Customization Guide* for information about specifying suffixes for BBISSP00.
- BBPARM member BBISSP00 has a new parameter, ALRTRCVE=YES | NO (where NO is the default). ALRTRCVE specifies whether this BBI-SS PAS is to be considered a product ALERT receiver. If YES is specified, the PAS receives ALERTs and alarm messages from other BMC Software products. If NO is specified, the PAS receives just ALERTs from other BMC Software products. The PAS can, however, process ALERTs created by MAINVIEW AutoOPERATOR EXECs, Rules, or AOAnywhere calls in this BBI-SS PAS. Refer to the *MAINVIEW Common Customization Guide* for information about parameters set in BBPARM member BBISSP00.
- BBPARM member BBISSP00 has a new parameter, XCFGROUP=xxxxxxx where xxxxxxxx is a user-specified one- to eight- character XCF group name.

**Note:** The MAINVIEW AutoOPERATOR BBI-PASs that are in different XCF Groups are unable to communicate with each other by using AOAnywhere. If you want to prevent *test* AOAnywhere callers from accessing *production* MAINVIEW AutoOPERATORs, SAF security is a better solution than using XCFGROUP.

Also note that the BBISSPxx ALRTRCV=YES setting is incompatible with the XCFGROUP parameter. Do not specify the XCFGROUP parameter in any BBI-PAS that has ALRTRCV=YES.

- BBPARM member BBISSP00 has a new parameter, JESCNCT=YES | NO that enables you to use JES functions before JES is started or even if the BBI-SS PAS is started with SUB=MSTR specified. When JESCNCT=YES is specified, you can use the IMFEXEC JESALLOC and JESSUBM commands to allocate and submit JES jobs.

## What Are the Changes Related to IMS

The following changes were made to IMS regions in MAINVIEW AutoOPERATOR 6.3.01.

The following region types have been added to the Region Applications:

**JBP** non-message-driven IMS JAVA applications region

**JMP** message-driven IMS JAVA applications region

These region types support the new Java address spaces in IMS 8.1. For more information, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

A new status has been added to the Fast Path Data Entry Databases (DEDB):

**NR** Indicates the database is nonrecoverable. Note that this status is displayed only for nonrecoverable DEDBs that have open areas.

## What Are the Changes Related to CICS

A new IMFEXEC command, IMFEXEC CICS ALTER DUMPCODE command has been added, which specifies whether to secure the alteration of CICS SYSTEM and TRANSACTION DUMPCODES. Refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for more information about the syntax of this command.

## What Else Has Changed

In addition, MAINVIEW AutoOPERATOR version 6.3.01 contains the following changes:

- Additional IMFCC return codes for the IMFEXEC ALERT FUNCTION keywords have been added. For more information, refer to “Return Codes for FUNCTION Keywords” in the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

- QAODUMP, a new distributed utility EXEC, has been added. For more information, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.
- BBSAMP member QAOSMTP1 has been updated to make modifications simpler. This sample EXEC shows how to send an e-mail message through a SMTP mail server and is documented in the *MAINVIEW AutoOPERATOR Solutions Guide*. You should review QAOSMTP1 to determine whether you prefer to implement your current version of the QAOSMTP1 EXEC or upgrade to version 6.3.01.

After you apply PTFs BPO5552, BPO5580 and BPO5563, version 6.3.01 will support JES job numbers greater than 99,999 for JES2 and JES3 jobs. For more information, refer to “IMFJNUM Option and Seven-Digit JES Job Numbers (z/OS v1.2 and Later)” on page 19-6.

## How to Secure MAINVIEW AutoOPERATOR 6.3.01

As with MAINVIEW AutoOPERATOR 6.1.00, you can implement security for MAINVIEW AutoOPERATOR 6.3.01 with the external security manager (ESM) that is installed at your site. While you may continue to use BMC Software proprietary security (documented in *Implementing Security for MAINVIEW Products*), you might also choose to control security of MAINVIEW AutoOPERATOR through your ESM.

This release includes new or additional security for the CSM application and the AOAnywhere AOEXEC statements. Refer to the *Implementing Security for MAINVIEW Products* for more information.

Some BBPARM members contain EXECs that can aid in the definition and permission of MAINVIEW AutoOPERATOR resources. For the names and functions of these EXECs, refer to “Sample Members for MAINVIEW AutoOPERATOR Only” on page F-1.

## How Much CPU Does the New Release Use

The new release does not use a significantly higher amount of CPU than the previous release.

## Does MAINVIEW AutoOPERATOR 6.3.01 Use Common Storage Differently

MAINVIEW AutoOPERATOR 6.3.01 does not use common storage differently from MAINVIEW AutoOPERATOR 6.1.00.

## Does MAINVIEW AutoOPERATOR 6.3.01 Use Private Storage Differently

MAINVIEW AutoOPERATOR 6.3.01 does not use private storage differently from MAINVIEW AutoOPERATOR 6.1.00.

## Does This Version Communicate with Version 6.1.00

Any BBI-SS PAS with an activated MAINVIEW AutoOPERATOR 6.3.01 product can schedule an ALERT or EXEC to any MAINVIEW AutoOPERATOR 6.1.00 BBI-SS PAS. In addition, a MAINVIEW AutoOPERATOR 6.3.01 terminal session can access MAINVIEW AutoOPERATOR 6.1.00 panels.

You can install MAINVIEW AutoOPERATOR 6.3.01 on your test system and use it to review production operation until you are satisfied with the product's stability and functionality. Then you can migrate MAINVIEW AutoOPERATOR 6.3.01 through your other systems, one at a time, at a pace that is acceptable to your staff.

**Note:** Be sure to review compatibility restrictions in "What Has Changed about Rules" on page 3-5.

## Does Version 6.1.00 Communicate with This Version

ALERTs, EXECs, and Rules can be sent from a MAINVIEW AutoOPERATOR 6.1.00 BBI-SS PAS to a MAINVIEW AutoOPERATOR 6.3.01 BBI-SS PAS.

## Which IBM Software Does MAINVIEW AutoOPERATOR 6.3.01 Require

Refer to Chapter 1, “Installation Prerequisites,” in the *MAINVIEW Installation Requirements Guide* for information about the IBM software (and levels) that is required to run MAINVIEW AutoOPERATOR 6.3.01.

## What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01

The following BMC Software products are supported by MAINVIEW AutoOPERATOR 6.3.01 at these version levels or later:

- SYSPROG Services version 3.2.01
- MAINVIEW for CICS version 5.4.00
- MAINVIEW for IMS version 3.3.20
- MAINVIEW for DBCTL version 3.3.20
- MAINVIEW for OS/390 version 2.6.00

## What Is Not Supported by Version 6.3.01

MAINVIEW AutoOPERATOR 6.3.01 does not support any IBM software release that is not supported by IBM. This list includes

- IMS/ESA release earlier than version 7.0
- CICS Transaction Server release earlier than version 1.3
- MQSeries release earlier than version 1.2
- NetView versions 1, 2, 3
- OS/390 releases earlier than version 2.10
- Tivoli NetView for z/OS releases earlier than version 5.1

MAINVIEW AutoOPERATOR 6.3.01 does not support

- COR (CommandMQ On Ramp)
- STRATEGY=QUALIFIED Rules

Within the MAINVIEW AutoOPERATOR product, support for the IMFEXEC CMD (schedule an EXEC) has been dropped. This function has been replaced by the IMFEXEC SELECT(execname) command. For more information about IMFEXEC SELECT(execname), refer to the chapter “Using the IMFEXEC Statements” in either the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

## **What Functions Will Not Be Supported in the Next Release of MAINVIEW AutoOPERATOR**

Support for communicating with PATROL Enterprise Manager using a VTAM printer (see BBPARM member AAOALTxx) will be removed in the next release. This communication method is used to send AutoOPERATOR Alerts to PATROL EM. An I/O Concepts X-Direct coaxial gateway is required for PATROL EM to emulate a VTAM printer.

The more robust TCP/IP-based communication interface with PATROL EM is the recommended method (see Chapter 16, “Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM”) for synchronizing AutoOPERATOR Alerts with PATROL EM. This method also removes the requirement for the I/O Concepts hardware.

## **What Functions Will Not Be Supported in Future Releases of MAINVIEW AutoOPERATOR**

In a future release, BMC Software plans to discontinue use of documented variables by the Continuous State Manager application

When these documented variables are discontinued, BMC Software will attempt to identify all customer implementations that are dependent on these variables to provide an alternate method of functionality.

# Installation Considerations for MAINVIEW AutoOPERATOR 6.3.01

This section contains technique-specific installation considerations for migrating from previous releases of MAINVIEW AutoOPERATOR to MAINVIEW AutoOPERATOR version 6.3.01.

## Getting New BBKEYS

If a new release of MAINVIEW AutoOPERATOR is installed on the same CPU where a currently licensed copy resides, the new release uses the same set of BBKEYS as the current copy. You do *not* need to request a new set of BBKEYS. You are required to obtain a new set of BBKEYS only when changing or upgrading CPUs.

Refer to the *MAINVIEW Common Customization Guide* for more information about acquiring BBKEYS.

## Defining MAINVIEW AutoOPERATOR Resources to CICS

If you have customized your CICS regions for use with a previously released version of MAINVIEW AutoOPERATOR, the CICS resource definitions for your BMC Software products must be refreshed.

This update is accomplished by customizing and executing the BMC Software job CMRCSDES in the BBSAMP data set.

No other changes are needed except changing the CICS execution JCL to reflect the names of the new BBPARM data sets.

## Migrating by Using the OS/390 and z/OS Installer: Standard Installation

The *OS/390 and z/OS Installer Guide* refers to two installation methods:

- Standard installation provides a fast IEBCOPY installation process. This method offers an optional step that creates a predefined SMP/E environment for SMP/E maintenance. The Standard installation method requires less expertise than an SMP/E installation, but it offers less control over the SMP/E environment.

- SMP/E is an industry-standard installation program that provides a variety of capabilities for unloading products and maintenance. SMP/E provides more flexibility than the Standard installation method for setting up and using an SMP/E environment, but it is more complicated and time consuming.

If you are migrating by using standard installation with the BMC Software OS/390 and z/OS Installer, read the following section on implementing the JES2MAO exit again.

If you are replacing any MAINVIEW AutoOPERATOR release and *all* of the following statements are true, you must implement the JES2MAO exit again:

- Your site is licensed for the MAINVIEW AutoOPERATOR for OS/390 option.
- You elected to implement the JES2MAO exit.
- The JES2MAO exit is linked to the existing system's BBLINK library.

Refer to “Displaying MVS JES2 Job Numbers in Address Space Application” on page 19-4, for additional information about the JES2MAO exit.

## AutoCustomization Considerations When Replacing Existing Products

When replacing existing BMC Software products, the following conditions apply:

- All existing products that are replaced must be customized again.
- Any modifications or additions to the UBBPARM data set that were performed manually after running AutoCustomization on the existing system need to be performed as necessary on the new UBBPARM data set after it is created by AutoCustomization on the new system.

Refer to “Customize the Installed Products” in the *OS/390 and z/OS Installer Guide* for additional information about AutoCustomization.

## **Migrating by Using System Maintenance Program (SMP)**

Read this section if you are using the System Maintenance Program (SMP) to migrate.

If MAINVIEW AutoOPERATOR 6.3.01 is installed into a set of SMP zones that contains any release of the BMC Software products listed in “What Other BMC Software Products Run with MAINVIEW AutoOPERATOR 6.3.01” on page 3-12, the minimum release level of those products must also be installed.

---

---

# Chapter 4      Continuous Operation: Using the Dynamic Parameter Manager

The Dynamic Parameter Manager (DPM) is a panel-driven application where you can view or modify parameters in BBPARM members from a TS session. The BBPARM members currently supported by DPM are as follows:

<b>AAOALSxx</b>	MAINVIEW AutoOPERATOR ALERT thresholds
<b>AAOARPxx</b>	MAINVIEW AutoOPERATOR Automation Reporter parameters
<b>AAOEXPxx</b>	MAINVIEW AutoOPERATOR EXEC parameters
<b>AAOTSPxx</b>	TapeSHARE for MAINVIEW AutoOPERATOR parameters

Additionally, the DPM application has an Activate command that can be used to dynamically change the BBPARM members that a BBI-SS PAS currently uses *without restarting the BBI-SS PAS*.

## Tasks That You Can Accomplish with the Dynamic Parameter Manager

With the Dynamic Parameter Manager, you can

- control virtual storage consumption

Use DPM to dynamically adjust the virtual storage throttles in AAOEXPxx and AAOALSxx (where xx is a user-defined suffix).

- control CPU & I/O usage

Use DPM to adjust the CPU & I/O throttles dynamically in AAOEXP $xx$ .

- implement previously defined AAOALS $xx$ , AAOARP $xx$ , AAOEXP $xx$ , and AAOTSP $xx$  members dynamically

By implementing previously defined members dynamically, different resource thresholds can be defined in response to anticipated environmental requirements without restarting the BBI-SS PAS.

- activate and control the data collection time intervals for the Automation Reporter application

The Automation Reporter collects data for specific automation items and the AAOARP $xx$  member controls how often data is collected and offloaded into the database.

The Automation Reporter application (and how to activate it with the Dynamic Parameter Manager) is described in the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

- customize the way the TapeSHARE for MAINVIEW AutoOPERATOR component performs tape device sharing automation among your OS/390 images

With MAINVIEW AutoOPERATOR TapeSHARE, you can automate the sharing of tape devices between OS/390 images. For more information about installing TapeSHARE, refer to “Implementing TapeSHARE” on page 6-1; for more information about using TapeSHARE, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

## Accessing the Dynamic Parameter Manager

Choose Option 9, Dynamic Parameter Manager, from the PRIMARY OPTION MENU (shown in Figure 4-1 on page 4-3) to see a list of the AAOALS $xx$ , AAOARP $xx$ , AAOEXP $xx$ , and AAOTSP $xx$  members in BBPARM.

**Figure 4-1 MAINVIEW AutoOPERATOR PRIMARY OPTION MENU**

```
BMC Software ----- PRIMARY OPTION MENU-----MAINVIEW AutoOPERATOR
OPTION ==>>>                                     DATE -- 01/02/01
                                                    TIME -- 12:22:53
                                                    USERID -- BAOMXY2
                                                    MODE -- ISPF 4.8

Operator Workstations:
  1 ALERT Management                               5 NetView Resources
  2 MVS Resources                                  6 TapeSHARE
  3 CICS Resources                                 7 MQSeries
  4 IMS Resources

Automation:
  8 Basic and Advanced Automation                 9 Dynamic Parameter Manager

General Services:
  C Service Refresh Cycle Setup                   K Current PF Key Assignments
  L Display Journal                               T Tutorials
  M Display Messages and Codes                     X Terminate

                                                    PF1/13 HELP PF3/15: EXIT

Copyright 2002, BMC Software, Inc. All rights reserved.
```

When you select Option 9, the Dynamic Parameter Manager panel is shown (Figure 4-2 on page 4-4).

**Figure 4-2 Dynamic Parameter Manager—Example 1**

```

BMC Software ----- Dynamic Parameter Manager ----- MAINVIEW AutoOPERATOR
COMMAND ==>>>                                     TGT ==>>  SYSB
Primary command: Add                                Date --- 01/02/09
                                                    Time --- 11:41:36

LC CMDS --- (S)elect, (A)ctivate

Cmd   Member          Lib VV.MM   Created      Changed      Size  ID
-----
-   AAOALS00          1  01.11   01/01/08   01/01/28 20:12      4   JDB1
-   AAOALS01          1  00.01   01/01/08   01/01/08 13:30      3   JDB1
-   AAOARP01          1  00.01   01/01/08   01/01/08 13:30      3   JDB1
-   AAOEXP00          1  02.26   01/01/12   01/01/28 20:12     27   JDB1
-   AAOTSP00          1  02.01   01/01/12   01/01/28 20:12     12   JDB1
***** END OF MEMBERS *****
    
```

Use this panel to see a list of all the AAOALS $_{xx}$ , AAOARP $_{xx}$ , AAOEXP $_{xx}$ , and AAOTSP $_{xx}$  members in BBPARM where the

- AAOALS $_{xx}$  members contain parameters that determine how much extended storage MAINVIEW AutoOPERATOR ALERTs use on your system
- AAOARP $_{xx}$  members contain parameters that control how the Automation Reporter application operates
- AAOEXP $_{xx}$  members contain parameters and thresholds that determine how efficiently MAINVIEW AutoOPERATOR EXECs execute on your system
- The AAOTSP $_{xx}$  members contain parameters that control how TapeSHARE operates

From this panel, you can

- use the line command Select in the **Cmd** column to select a member to view and modify the parameters, if you choose

For more information about selecting an AAOALS $_{xx}$  member, refer to “Modifying ALERT Thresholds in AAOALS $_{xx}$ ” on page 4-7.

For more information about selecting an AAOARP<sub>xx</sub> member, refer to “Modifying Automation Reporter Application Parameters in AAOARP<sub>xx</sub>” on page 4-13.

For more information about selecting an AAOEXP<sub>xx</sub> member, refer to “Modifying EXEC Parameters in AAOEXP<sub>xx</sub>” on page 4-17.

For more information about selecting an AAOTSP<sub>xx</sub> member, refer to “Modifying TapeSHARE for MAINVIEW AutoOPERATOR Parameters in AAOTSP<sub>xx</sub>” on page 4-30.

- use the line command `Activate` with the member to either
  - dynamically change the `BBPARAM` member that your system is currently using
  - implement any changes you might have made to a member

The `Activate` command displays the Confirm Parameter Activation panel. This function allows you to verify that you want to dynamically change an AAOALS<sub>xx</sub>, AAOARP<sub>xx</sub>, AAOEXP<sub>xx</sub>, or AAOTSP<sub>xx</sub> member.

For more information about activating your changes, refer to “Dynamically Implementing New Parameter Settings with the `Activate` Command” on page 4-38.

- use the primary command `ADD` to add a new AAOALS<sub>xx</sub>, AAOARP<sub>xx</sub>, AAOEXP<sub>xx</sub>, or AAOTSP<sub>xx</sub> member. To add new members, type one of the following commands:

**ADD AAOALS<sub>xx</sub>**

**ADD AAOARP<sub>xx</sub>**

**ADD AAOEXP<sub>xx</sub>**

**ADD AAOTSP<sub>xx</sub>**

on the `COMMAND` line, where *xx* is a new suffix for a new member. For example, if you want to add a new member named AAOEXP02, type

**ADD AAOEXP02**

## Describing the Fields

The fields for the Dynamic Parameter Manager panel are as follows:

Field	Description
Cmd	enables you to enter the Select or Activating line command
Member	is the list of members that can be  AAOALS $_{xx}$ for ALERTs thresholds  AAOARP $_{xx}$ for Automation Reporter operation  AAOEXP $_{xx}$ for EXEC thresholds  AAOTSP $_{xx}$ for TapeSHARE operation
Lib	is the library volume number
VV.MM	indicates the version level (VV) and modification level (MM)
Created	is the date the member was created
Changed	is the date and time the member was last edited
Size	is the size (in number of lines) of the member
ID	is the user ID of the person who last changed the member or, if the member was last changed with the Dynamic Parameter Managers, the job name of the subsystem used

## Filtering the Display of Data

You can qualify (or filter) the members shown on the panel by entering a prefix or string of characters in the input field located under the column heading **Member**. For example, to see all the member names of AAOEXP $_{xx}$ , enter the characters **AAOEXP** in the input field:

⋮	Cmd	Member	Lib	VV.MM	Created	Changed	Size	ID
		AAOEXP						
		-----						
⋮								

The output shows only those members with the prefix AAOEXP.

# Modifying ALERT Thresholds in AAOALSxx

BBPARM members AAOALSxx contain the parameters that determine how much storage MAINVIEW AutoOPERATOR ALERTs can use on your system.

For Information About	See Page
Activating and modifying the Automation Reporter application	4-13
Modifying values for MAINVIEW AutoOPERATOR EXECs	4-17
Customizing TapeSHARE	4-30

To modify an AAOALSxx member, type S in the **Cmd** field of the Dynamic Parameter Manager panel. Figure 4-3 shows an example of the panel.

**Figure 4-3 Dynamic Parameter Manager Panel—Example 2**

```

BMC Software ----- Dynamic Parameter Manager ----- AutoOPERATOR
COMMAND ==>
Primary command: Add
Date --- 01/02/19
Time --- 11:41:36

LC CMDS --- (S)elect, (A)ctivate

Cmd  Member          Lib VV.MM  Created      Changed      Size  ID
-----
s    AAOALS00          1  01.11  01/03/08  01/03/28  20:12    4   JDB1
_    AAOALS01          1  00.01  01/03/08  01/03/08  13:30    3   JDB1
_    AAOARP01          1  00.01  01/01/08  01/01/08  13:30    3   JDB1
_    AAOEXP00          1  02.26  01/05/12  01/03/28  20:12   27   JDB1
_    AAOTSP00          1  02.01  01/07/12  01/03/28  20:12   12   JDB1
***** END OF MEMBERS *****
    
```

Use this panel to see a list of all the AAOALSxx, AAOARPxx, AAOEXPxx, and AAOTSPxx members in BBPARM where the

- AAOALSxx members contain parameters that determine how much extended storage MAINVIEW AutoOPERATOR ALERTs use on your system
- AAOARPxx members contain parameters that control how the Automation Reporter application operates

- AAOEXPxx members contain parameters and thresholds that determine how efficiently MAINVIEW AutoOPERATOR EXECs execute on your system
- AAOTSPxx members contain parameters that control how TapeSHARE operates

From this panel, you can

- use the Select line command in the **Cmd** column to select a member to view and modify the parameters, if you choose
- use the Activate line command when you want to dynamically change the BBPARM member that your system is currently using to manage EXEC thresholds, ALERTs storage, or the Automation Reporter or TapeSHARE applications

The Activate command displays the Confirm Parameter Activation panel; see Figure 4-13 on page 4-39. This function allows you to verify that you want to dynamically change and use a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOTSPxx member.

For more information about activating your changes, refer to “Dynamically Implementing New Parameter Settings with the Activate Command” on page 4-38.

- use the primary command ADD to add a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOTSPxx member

To add a new member or members, type one of the following commands on the COMMAND line, where xx is a new suffix for a new member:

**ADD AAOALSxx**

**ADD AAOARPxx**

**ADD AAOEXPxx**

**ADD AAOTSPxx**

For example, if you want to add a new member named AAOALS02, type

**ADD AAOALS02**

## Setting ALERTs Storage Thresholds

Use the Alerts Storage Thresholds panel to see and modify thresholds that control how much storage ALERTs can use on your system.

Figure 4-4 shows an example of this panel.

**Figure 4-4**      **ALERTs Storage Thresholds Panel**

```

BMC Software ----- Alerts Storage Thresholds ----- AutoOPERATOR
COMMAND ==>>>                                     TGT --- BBSYSB
                                                    DATE --- 01/11/08
                                                    TIME --- 17:54:58

                Member                Current
MAXSTOR ==>>>   1000                    1000 Maximum storage (ext private) in K bytes
WARNLVL1 ==>>>    60                      60 Initial threshold (percent)
WARNLVL2 ==>>>    75                      75 Secondary threshold (percent)
ALERTNV ==>>>    NO                       NO Default RETAIN setting for Alerts which
                                                    have no explicit RETAIN setting.
                                                    Requires a Warm Start for change to be
                                                    effective.
Publish ==>>>    ADD                      ADD Default PUBLISH setting for Alerts. Used
                                                    for Alerts which have no explicit PUBLISH
                                                    setting.

Press END to save changes, CANCEL to cancel changes

```

This panel provides two columns of data:

- on the left, the value of the parameters in the AAOALSxx member for the target BBI-SS PAS

You can enter your modifications on this side of the panel.

- on the right, the value of the parameters that are currently in effect for the target BBI-SS PAS

These values cannot be edited.

## Describing the Fields

Descriptions of the fields on this panel follow:

<b>Field Name</b>	<b>Description</b>
MAXSTOR	maximum amount of BBI-SS PAS extended private virtual storage to be allocated to ALERTs

Specify the storage amount in kilobytes and thousandths of megabytes. The default is zero, which means *no checking is done*.

WARNLVL1	first warning threshold level for queued ALERTs
----------	---

The default is 60%, which means that if MAXSTOR=1K, a warning message is issued when the amount of virtual storage allocated to queued ALERTs reaches .6K. The message looks like the following message:

```
AU6100W ALERT STORAGE USAGE 60% OVER 1ST LEVEL (60%)
```

WARNLVL2	second warning threshold level for queued ALERTs
----------	--

The default is 75%, which means that if MAXSTOR=1K, a warning message is issued when the amount of virtual storage allocated to queued ALERTs reaches .75K. The message looks like the following message:

```
AU6100W ALERT STORAGE USAGE 75% OVER 2ND LEVEL (75%)
```

When the maximum amount of storage is used (as set by MAXSTOR), two additional messages are produced. The first message is

```
EM0022E ERROR PROCESSING .. DEMOAL22 .. ALERT NOT
      ADDED QUEUE IS FULL
```

where DEMOAL22 is the name of the ALERT that was not added to the ALERT queue because the maximum storage limit has been reached.

The second message is

```
EM0020E - IMFEXEC ALERT DASD5.11:57:28 'RESERVE IN
      PROGRESS...
```

where the text of the message is the text of the IMFEXEC ALERT command that attempted to store the ALERT

To solve this problem, use a Rule to interrogate the information fields in the message and delete less important ALERTs.

No more ALERTs are queued until the amount of storage used by queued ALERTs drops below MAXSTOR. When the shortage condition is alleviated, the following message is issued:

```
AU6200I ALERT STORAGE USAGE 58%, NO LONGER OVER  
THRESHOLD
```

ALERTNV default setting for the RETAIN parameter of IMFEXEC ALERT statements

The default is NO. Specifying YES changes the default of all generated ALERTs to IMFEXEC ALERT...RETAIN(YES), which means that all generated ALERTs are retained in disk storage across BBI-SS PAS restarts and OS/390 IPLs.

Specifying NO means that all ALERTs are not retained across BBI-SS PAS restarts or OS/390 IPLs. When NO is specified in AAOALSxx, you can still override the setting and create nonvolatile ALERTs by specifying RETAIN(YES) on a specific IMFEXEC ALERT statement. When you override this setting, only that ALERT are retained across BBI-SS PAS restarts and OS/390 IPLs.

**Note:** When an ALERT is targeted to a different BBI-SS PAS, by default, the ALERT retains the ALERTNV setting of the target BBI-SS PAS.

When you change the setting on the ALERTNV parameter, the BBI-SS PAS must be warm-started for the change to take effect.

**PUBLISH**      whether MAINVIEW AutoOPERATOR should (by default) publish ALERTs to PATROL EM consoles that are attached through GME

Possible values are as follows:

**ADD:** publish, but do not replace a previously published ALERT

**REPLACE:** publish, but delete a previously published ALERT before adding the new one.

**NO:** do not publish ALERTs

The default is ADD.

# Modifying Automation Reporter Application Parameters in AAOARPxx

BBPARM members AAOARPxx contain the parameters that activate and control the data collection and offloading functions of the Automation Reporter application.

To modify an AAOARPxx member, type **S** in the **Cmd** field of the Dynamic Parameter Manager panel. Figure 4-5 on page 4-13 shows an example of the panel.

**Figure 4-5 Dynamic Parameter Manager Panel—Example 3**

```

BMC Software ----- Dynamic Parameter Manager ----- AutoOPERATOR
COMMAND ==>>>                                     TGT ==>> BBSYSA
Primary command: Add                                 Date --- 01/02/19
                                                    Time --- 11:41:36

LC CMDS --- (S)elect, (A)ctivate

Cmd  Member          Lib VV.MM   Created      Changed      Size  ID
---  -
-    AAOALS00         1  01.11   01/03/08   01/03/28 20:12    4   JDB1
-    AAOALS01         1  00.01   01/03/08   01/03/08 13:30    3   JDB1
-    AAOARP01         1  00.01   01/01/08   01/01/08 13:30    3   JDB1
-    AAOEXP00         1  02.26   01/05/12   01/03/28 20:12   27   JDB1
-    AAOTSP00         1  02.01   01/07/12   01/03/28 20:12   12   JDB1
***** END OF MEMBERS *****
    
```

Use this panel to see a list of all the AAOALSxx, AAOARPxx, AAOEXPxx, and AAOTSPxx members in BBPARM where the

- AAOALSxx members contain parameters that determine how much extended storage MAINVIEW AutoOPERATOR ALERTs use on your system
- AAOARPxx members contain parameters that control how the Automation Reporter application operates
- AAOEXPxx members contain parameters and thresholds that determine how efficiently MAINVIEW AutoOPERATOR EXECs execute on your system

- AAOOTSPxx members contain parameters that control how TapeSHARE operates

From this panel, you can

- use the line command **Select** in the **Cmd** column to select a member to view and modify the parameters, if you choose
- use the line command **Activate** when you want to dynamically change the BBPARM member that your system is currently using to manage EXEC thresholds, ALERTs storage, or the Automation Reporter or TapeSHARE applications

The **Activate** command displays the **Confirm Parameter Activation** panel; see Figure 4-13 on page 4-39. With this function you can verify that you want to dynamically change and use a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOOTSPxx member.

For more information about activating your changes, refer to “Dynamically Implementing New Parameter Settings with the **Activate** Command” on page 4-38.

- use the primary command **ADD** to add a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOOTSPxx member

To add a new member or members, type one of the following commands:

**ADD AAOALSxx**

**ADD AAOARPxx**

**ADD AAOEXPxx**

**ADD AAOOTSPxx**

on the **COMMAND** line, where **xx** is a new suffix for a new member. For example, if you want to add a new member named AAOARP02, type

**ADD AAOARP02**

## Setting Automation Reporter Parameters

Use the Automation Reporter Parms panel to

- see and modify parameters for the Automation Reporter application
- control how often data is collected
- name a data set to hold the data
- indicate when the collected data is offloaded to the data set

Figure 4-6 on page 4-15 shows an example of this panel.

**Figure 4-6 Automation Reporter Parms Panel**

```

BMC Software ----- Automation Reporter Parms ----- AutoOPERATOR
COMMAND ==>>>                                     TGT --- MS@1
                                                    DATE --- 01/02/28
                                                    TIME --- 13:39:59

                Member                Current

New DSN ==>>>
                Current ===

Disp    ==>>>      MOD                Dataset disposition (SHR, MOD)

Interval ==>>>      15                Interval minutes (10 - 1440, or 0)

Off int ==>>>      1440              Offload interval minutes (60 - 1440)

Alert Queue ==>>>  MAIN                Alert Queue

Press END to save changes, CANCEL to cancel changes
    
```

This panel provides two columns of data:

- on the left, the value of the parameters in the AAOARPx member for the target BBI-SS PAS

You can enter your modifications on this side of the panel.

- on the right, the value of the parameters that are currently in effect for the target BBI-SS PAS

These values cannot be edited.

The first time you bring up the Automation Reporter, no current data set values are available. Therefore, the **Current** === field and all the values on the right side of the panel are blank.

After you have used the Automation Reporter once, the next time you access this panel, the **Current** === field and all the display fields show the values you previously entered.

The left side of the panel displays the default values in BBPARM member AAOARPx, as shown in this panel.

### Describing the Fields

Descriptions of the fields on this panel follow:

<b>Field Name</b>	<b>Description</b>
New DSN	<p>Specify the name of the sequential data set that you have preallocated.</p> <p>The user-defined data set should be a fixed-block, sequential data set with a maximum record length of 580 bytes. This data set must be preallocated before the Automation Reporter can be activated.</p>
Disp	<p>Specify the disposition of the user-defined sequential data set.</p> <p>Valid disposition values are SHR (share) and MOD (modify), which is the default value. Specifying SHR causes the Automation Reporter to overwrite existing data. Specifying MOD causes the Automation Reporter to append data to the end of the data set from each offload cycle.</p>
Interval	<p>Specify the collection interval (in minutes) for the Automation Reporter to collect data.</p> <p>For example, if you specify 60, the Automation Reporter records the data in the subsystem for the resources and activities every 60 minutes.</p> <p>Valid collection intervals range from 10 minutes to 1440 minutes. The default is 15 minutes.</p>
Off Int	<p>Specify the offload interval (in minutes) for the Automation Reporter to offload the data it has collected to the data set.</p>

For example, if you specify 120, the Automation Reporter writes the data it has collected to the sequential data set every 2 hours.

Valid offloading intervals are 60 minutes to 1440 minutes. The default is 1440 (or once a day).

**Alert Queue** Specify the name of a MAINVIEW AutoOPERATOR ALERT queue to which any MAINVIEW AutoOPERATOR ALERTs created by the Automation Reporter will be queued.

For example, if you specify ARALRT, ALERTs are queued to a queue named ARALRT.

Any valid MAINVIEW AutoOPERATOR ALERT queue name can be entered in this field (default queue name is MAIN). You do not have to enter the name of a pre-existing ALERT queue. You can enter the name of a new queue and it is created for you when the Automation Reporter is activated and ALERTs are generated.

## Modifying EXEC Parameters in AAOEXPxx

BBPARM member AAOEXPxx contains parameters that determine the resource throttle settings and the environment for MAINVIEW AutoOPERATOR EXECs.

To modify an AAOEXPxx member, type **S** in the **Cmd** field next to a member on the Dynamic Parameter Manager panel (refer to Figure 4-2 on page 4-4).

Figure 4-7 on page 4-18 shows an example of the panel that is displayed when you select an AAOEXPxx member called AAOEXP00.

**Figure 4-7 Dynamic Parameter Manager Member Panel**

```

BMC Software ----- Dynamic Parameter Manager Member ----- AutoOPERATOR
COMMAND ==>>                                     TGT --- BBSYSA
                                                    Date --- 01/02/19
                                                    Time --- 11:41:44

                Member Name: AAOEXP00 - AutoOPERATOR EXEC Specifications

LC CMDS --- (S)elect

Cmd   Panel Description
-     EXEC Thresholds
-     EXEC Parameters
    
```

Use this panel to select additional panels that allow you to see and modify either

- EXEC Thresholds

You can set limits for how all your EXECs execute on your system; for example:

- the maximum number of EXECs that can run in the Normal or High priority queue
- the maximum number of EXECs that are allowed to abend before the EXEC Manager is terminated

For more information about selecting EXEC thresholds, refer to “Setting EXEC Thresholds” on page 4-19.

- EXEC Parameters

You can specify parameters for your EXECs themselves; for example:

- whether EXEC parameters are translated to uppercase
- what the maximum number of OSPI ACBs are

For more information about selecting EXEC parameters, refer to “Setting EXEC Parameters” on page 4-25.

## Setting EXEC Thresholds

Use the EXEC Thresholds panel to view or modify values that determine how many EXECs can run in your system in the Normal or high-priority queues, how many EXEC abends that you want to be able to support, and so on.

Figure 4-8 shows an example of the EXEC Thresholds panel.

**Figure 4-8 EXEC Thresholds Panel**

```

BMC Software ----- EXEC Thresholds ----- AutoOPERATOR
COMMAND ==>>>                                     TGT --- BBSYSA
                                                    DATE --- 01/02/19
                                                    TIME --- 11:41:52

                Member                Current
EMABCNT ==>>>      5                   5   # of ABENDS allowed before EM terminated
EXABCNT ==>>>      3                   3   # of ABENDS allowed before EXEC disabled

MAXNORM ==>>>      5                   5   Maximum concurrent Normal EXECs
MAXHIGH ==>>>      3                   3   Maximum concurrent High Priority EXECs
MAXNORMQ ==>>>     5                   5   Maximum Normal EXECs Queued
MAXHIGHQ ==>>>     3                   3   Maximum High Priority EXECs Queued

WARNLVL1 ==>>>    65                   65   Initial threshold (percent)
WARNLVL2 ==>>>    80                   80   Secondary threshold (percent)

SELLIM  ==>>>     10                   10   Maximum depth for SELECT with WAIT(YES)
MAXTPUT ==>>>      0                   0   Maximum TPUT allowed in each EXEC

Press END to save changes,  CANCEL to cancel changes
    
```

This panel provides two columns of data:

- on the left, the value of the parameters in the AAOEXPxx member for the target BBI-SS PAS

You can enter your modifications on this side of the panel.

- on the right, the value of the parameters that are currently in effect for the target BBI-SS PAS

These values cannot be edited.

## Scheduling EXECs

With the EXEC Thresholds panel, you can specify the number of EXECs that can be scheduled to the High and Normal priority queues by using MAXNORM and MAXHIGH parameters. When the number of scheduled EXECs matches the values set in these parameters, *all scheduling stops* until the number of EXECs drops below the thresholds.

You can also specify two thresholds for the High and Normal queues that, when reached, issue warning messages telling you that the threshold has been reached. These thresholds are WARNLVL1 and WARNLVL2 and the values represent percentages of MAXNORMQ and MAXHIGHQ.

## Examples

For example, you can specify

**MAXNORMQ ==> 150**  
**MAXHIGHQ ==> 100**

and

**WARNLVL1 ==> 50**  
**WARNLVL2 ==> 75**

For the Normal queue, warning messages are sent when 75 EXECs are scheduled and again when 112 EXECs are scheduled.

WARNLVL1 50% OF 150 = 75  
 WARNLVL2 75% OF 150 = 112

For the High queue, warning messages are sent when 50 EXECs are scheduled and again when 75 EXECs are scheduled.

WARNLVL1 50% OF 100 = 50  
 WARNLVL2 75% OF 100 = 75

Scheduling of EXECs stops when the maximum values are reached. EXEC scheduling automatically resumes when the number of EXECs drops below the maximum. Refer to the descriptions of the MAXNORMQ, MAXHIGHQ, WARNLVL1, and WARNLVL2 fields in “Describing the Fields” on page 4-21 for more information.

## Describing the Fields

A description of the fields on this panel follows.

Field Name	Description
Member	specifies the values for the new AAOEXPxx member
Current	<p>specifies the values in the AAOEXPxx member that the BBI-SS PAS is currently using</p> <p>These values are stored in CSA.</p> <p>The member that the BBI-SS PAS is currently using is one of the following choices:</p> <ul style="list-style-type: none"> <li>— the AAOEXPxx member that was used during the most recent BBI-SS PAS initialization</li> <li>— the most recently applied AAOEXPxx member</li> </ul> <p>EMABCNT is the maximum number of abends the EXEC Manager can sustain before it shuts down and requires a restart of the BBI-SS PAS</p> <p>When the EXEC Manager stops, no EXECs can run in the system. Valid values range from 1 to 99. The default is 5.</p> <p>Use this threshold to avoid recursive or persistent abends that demand attention.</p> <p>Abends in individual EXECs do not count against this threshold. The count is cumulative from the time the target BBI-SS PAS was started. The reason is that generally, abends in individual EXECs are more frequent and are caused by coding errors within REXX or coding errors within an IMFEXEC statement. These abends usually affect only one or two EXECs or a set of similar EXECs. Multiple failures of this type are not serious enough to warrant shutting off all automation but are serious enough to warrant shutting off the failing EXECs.</p> <p>However, EXEC Manager abends are much less frequent and far more serious. They can usually be recovered from, but if several occur, it is better to stop MAINVIEW AutoOPERATOR and restart it to avoid further errors.</p>

**EXABCNT** is the maximum number of abends that a single EXEC can sustain before it is disabled

Valid values range from 1 to 99 and the default is 5. For example, if you specify EXABCNT=10 and an EXEC abends 10 times, the EXEC is disabled after the tenth abend.

**MAXNORM** is the maximum number of EXECs that can execute concurrently on Normal priority threads

Valid values range from 1 to 99 and the default is 1.

**MAXHIGH** is the maximum number of EXECs that can execute concurrently on high-priority threads

Valid values range from 1 to 99 and the default is 5.

**MAXNORMQ** is the maximum number of normal priority EXECs that may be queued for execution

Valid values range from 0 to 999 and the default is 0. A value of 0 means that *no checking is done*.

When the maximum number is reached, *no more EXECs are scheduled* to the Normal priority queue and a warning message is issued:

**BMC**EM6100W MAXNORMQ EXEC 19/ 20, OVER LAST LEVEL (95%)

EXEC scheduling resumes when the queue size drops below the maximum. An informational message is issued to inform you that scheduling has resumed:

**BMC**EM6200I MAXNORMQ EXEC 9/ 20, NO LONGER OVER THRESHOLD

**Note:** EXECs scheduled with WAIT=YES are never queued. They are executed on the thread of the EXEC that selected it.

**MAXHIGHQ** is the maximum number of High-priority EXECs that may be queued for execution

Valid values range from 0 to 999 and the default is 0. A value of 0 means that *no checking is done*.

When the maximum number is reached, *no more EXECs are scheduled* to the high-priority queue and a warning message is issued:

**BMC**EM6100W **MAXHIGHQ EXEC 19/ 20,OVER LAST LEVEL (95%)**

EXEC scheduling resumes when the queue size drops below the maximum. An informational message is issued to inform you that scheduling has resumed:

**BMC**EM6200I **MAXHIGHQ EXEC 9/ 20, NO LONGER OVER THRESHOLD**

**Note:** EXECs scheduled with the IMFEXEC command WAIT=YES are never queued. They are executed on the thread of the EXEC that selected it.

**WARNLVL1** specifies a preliminary percentage of EXECs in a queue that, when met, generates a warning message

Valid values range from 0% to 99% and the default is 60%.

Use this threshold to set the first percentage that triggers a warning message to be sent. For example, if you set MAXHIGHQ=100, and WARNLVL1=60, a warning message is issued when 60 EXECs are queued to the High-priority queue. The message looks like the following example:

**BMC**EM6100W **MAXHIGHQ EXEC 60/ 100, OVER 1ST LEVEL (60%)**

This threshold is applicable to the High and Normal priority queues.

**WARNLVL2** specifies a secondary percentage of EXECs in a queue that, when met, generates a warning message

Valid values range from 0% to 99% and the default is 75%.

Use this threshold to set a second percentage that triggers a warning message to be sent.

For example, if you set MAXHIGHQ=100 and WARNLVL2=75, a warning message is issued when 75 EXECs are queued to the high-priority queue. The message looks like the following example:

**BMC**EM6100W **MAXHIGHQ EXEC 75/ 100, OVER 2ND LEVEL(75%)**

In addition, a second message is issued that identifies the first EXEC that was not scheduled because of this queuing condition:

**BMC**EM6150W EXEC-COLORS NOT QUEUED,OVER LAST LEVEL THRESHOLD

In this example, Colors is the name of an EXEC that was not scheduled. Additionally, no more EXECs are scheduled to the high-priority queue until the usage percentage of the queue is less than the value set for MAXHIGHQ. This threshold is applicable to the High and Normal priority queues.

SELLIM

is the maximum number of active EXECs on a single thread

Valid values range from 0 to 99 and the default is 0.

This parameter is designed to control recursive calls (where EXECA schedules EXECB, and EXECB schedules EXECC, and so on) that run out of control, and can potentially fill up the private storage in the BBI-SS PAS and cause automation to stop.

An example of recursive calls is EXECA schedules EXECB with the WAIT=YES parameter and EXECB in turn, calls EXECA.

MAXTPUT

is the maximum number of TSO TPUTs that is allowed for each execution of an EXEC

Valid values range from 0 to 999 and the default is 0. TPUTs for EXECs scheduled with the WAIT=YES parameter are counted separately from the scheduling EXEC. If the maximum is exceeded, the following message is issued:

**BMC**EM0026W TPUT JOURNALLING SUSPENDED,BACKLOG HAS EXCEEDED MAXIMUM

and subsequent TPUTs are ignored until the EXEC terminates.

## Setting EXEC Parameters

Use the EXEC Parameters panel to view or modify values that determine how individual EXECs execute in your system.

Figure 4-9 shows an example of the panel.

**Figure 4-9 EXEC Parameters Panel**

```

BMC Software ----- EXEC Parameters ----- AutoOPERATOR
COMMAND ==>>> TGT --- JO63
DATE --- 02/06/05
TIME --- 11:59:24

Member Current
TIMEXLIM ==>> 0 0 Maximum CPU time limit for an EXEC
PEREXLIM ==>> 0 0 Maximum CPU percentage usage for an EXEC
TSOTIME ==>> 9999 9999 Timeout value for attached TSO commands

UCPARMS ==>> YES YES Translate EXEC parms to UPPERCASE
AUDITMSG ==>> YES YES Write JRNL msg for BLDL and submit EXEC
OSPINUM ==>> 0 0 Maximum # of OSPI ACBs
OSPIPRFX ==>> OSPI OSPI 4 character OSPI ACB prefix
UNITNAME ==>> SYSALLDA SYSALLDA UNITNAME used for allocating a dataset
PREFIX ==>> JO63 JO63 Prefix used for allocation a dataset
SUBXAUTH ==>> RACF RACF RACF/JOBNAME/DEFER - IMFSUBEX security

EXEC ==>> (Enter ? to see high priority EXECs)

Press END to save changes, CANCEL to cancel changes

```

Use this panel to continue setting parameters for how single EXECs execute on the system. As on the EXEC Thresholds panel, this panel provides two columns of data:

- on the left, the value of the parameters in the AAOEXPxx member for the target BBI-SS PAS

You can enter your modifications on this side of the panel.

- on the right, the value of the parameters that are currently in effect for the target BBI-SS PAS

These values cannot be edited.

From this panel, you can specify a question mark (?) in the **EXEC ==>>** field to display another panel where you can specify the names of EXECs that will always execute on High priority threads.

Refer to “Setting High-Priority EXECs” on page 4-28 for more information about specifying EXEC names for high-priority queuing.

### Describing the Fields

Descriptions of the fields on this pane follow:

<b>Field Name</b>	<b>Description</b>
TIMEXLIM	<p>is the maximum CPU time limit allowed for an EXEC to use while executing</p> <p>Valid values range from 0 to 9999 and the default is 0.</p> <p>Set this parameter to ensure that any EXEC does not execute limitlessly.</p>
PEREXLIM	<p>is the maximum CPU percentage usage that is allowed for an EXEC to use while executing</p> <p>Valid values range from 0 to 99 and the default is 0.</p> <p>Set this parameter to ensure that any EXEC does not use a disproportionate amount of the CPU time.</p>
TSOTIME	<p>is the time-out value (in seconds) for attached TSO commands</p> <p>Valid values range from 0 to 9999 and the default is 1200.</p> <p>Set this parameter to limit the amount of time that a TSO command associated with an EXEC can use.</p>
UCPARMS	<p>specifies that parameters passed to an EXEC must be converted to uppercase characters</p> <p>Valid values are YES and NO and the default is YES. All CLIST EXECs require uppercase characters for parameters.</p>

AUDITMSG	specifies whether you want to record in the BBI-SS PAS Journal all occurrences of BLDLs against the SYSPROC data set and all user-initiated EXEC requests										
	Valid values are YES and NO and the default is YES.										
OSPINUM	specifies the maximum number of OSPI ACBs that is allowed										
	Valid values range from 0 to 999 and the default is 0.										
OSPIPRFX	specifies the default four-character OSPI ACB prefix										
	Valid values are any four-character string and the default is OSPI.										
UNITNAME	is the default UNITNAME that is used for allocating a data set										
	Valid values are any eight-character string and the default is SYSALLDA.										
PREFIX	is the prefix that is used for allocating a data set in an EXEC if the data set name is not enclosed in quotation marks										
	Valid values are any seven-character string and the default is blank, which refers to the subsystem ID.										
SUBXAUTH	specifies one of the following names:										
	<table> <thead> <tr> <th>Name</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>DEFER</td> <td>defers the use of the job name or user ID for the security check to the batch job</td> </tr> <tr> <td>RACF</td> <td>uses the RACF user ID for all batch job security checks</td> </tr> <tr> <td>USER ID</td> <td>uses the user ID for all batch job security checks</td> </tr> <tr> <td>JOBNAME</td> <td>uses the job name for all batch job security checks</td> </tr> </tbody> </table>	Name	Meaning	DEFER	defers the use of the job name or user ID for the security check to the batch job	RACF	uses the RACF user ID for all batch job security checks	USER ID	uses the user ID for all batch job security checks	JOBNAME	uses the job name for all batch job security checks
Name	Meaning										
DEFER	defers the use of the job name or user ID for the security check to the batch job										
RACF	uses the RACF user ID for all batch job security checks										
USER ID	uses the user ID for all batch job security checks										
JOBNAME	uses the job name for all batch job security checks										
	The default is DEFER.										

REXX shows the current REXX =setting

Valid values are YES or NO and the default is YES.

Restriction
The number of EXECs that can be scheduled concurrently is equal to $(x - 3) / 2$ where $x$ is the number of REBs. At BBI-SS PAS startup time, the following warning message is issued if the aggregate of MAXNORM and MAXHIGH exceeds this limit:  EM0002W EXEC MULTITASKING EXCEEDS SITE LIMITS  In addition, further scheduling of EXECs is inhibited if this limit is reached.

EXEC enables you to display another panel where you can define the names of EXECs that will always run on High priority threads

Refer to “Setting High-Priority EXECs” for more information about this panel.

## Setting High-Priority EXECs

To update the list of high-priority EXECs, enter a question mark (?) on the EXEC Parameters panel, next to the **EXEC** ==> field in the **Member** column. This displays the Update Non-Unique Parameter List panel.

Figure 4-10 on page 4-29 shows an example of the panel.

**Figure 4-10 Update Non-Unique Parameter List**

```
BMC Software ----- Update Non-Unique Parameter List ----- AutoOPERATOR
COMMAND ==>
Parameter: EXEC=      Enter ADD xxxx to add new entries      TGT --- BBSYSA
LC CMDS --- (D)delete                                     DATE --- 01/02/19
                                                         TIME --- 11:27:27

- EXEC1
- EXEC2
- EXEC3
- EXEC4
- EXECABCD
- EXECEFGH
***** END OF ENTRIES *****
```

To add an EXEC for high-priority execution, type

**ADD *execname***

where *execname* is the one- to eight-character name of the EXEC that you want to execute in the high-priority queue.

To delete an EXEC, use the line command Delete in the **Cmd** field next to the EXEC name that you want to delete.

# Modifying TapeSHARE for MAINVIEW AutoOPERATOR Parameters in AAOTSPxx

BBPARM members AAOTSPxx contain the parameters that you can modify to customize the way that the TapeSHARE for MAINVIEW AutoOPERATOR component automates the sharing of tape devices among your OS/390 images.

Remember, the TapeSHARE for MAINVIEW AutoOPERATOR component is designed to perform tape sharing automation without you needing to customize any of these parameters. For more information about TapeSHARE and how these parameters affect TapeSHARE, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

To modify an AAOTSPxx member, type **S** in the **Cmd** field of the Dynamic Parameter Manager panel.

Figure 4-11 shows an example of the panel.

**Figure 4-11 Dynamic Parameter Manager Panel—Example 4**

```

BMC Software ----- Dynamic Parameter Manager ----- AutoOPERATOR
COMMAND ==>> TGT ==>> BBSYSA
Primary command: Add Date --- 01/02/19
Time --- 11:41:36

LC CMDS --- (S)elect, (A)ctivate

Cmd  Member          Lib VV.MM  Created      Changed      Size  ID
-----
_   AAOALS00          1  01.11  01/03/08  01/03/28  20:12    4  JDB1
_   AAOALS01          1  00.01  01/03/08  01/03/08  13:30    3  JDB1
_   AAOARP01          1  00.01  01/01/08  01/01/08  13:30    3  JDB1
_   AAOEXP00          1  02.26  01/05/12  01/03/28  20:12   27  JDB1
S   AAOTSP00          1  02.01  01/07/12  01/03/28  20:12   12  JDB1
***** END OF MEMBERS *****
    
```

Use this panel to see a list of all the AAOALSxx, AAOARPxx, AAOEXPxx, and AAOTSPxx members in BBPARM where the

- AAOALSxx members contain parameters that determine how much extended storage MAINVIEW AutoOPERATOR ALERTs use on your system
- AAOARPxx members contain parameters that control how the Automation Reporter application operates
- AAOEXPxx members contain parameters and thresholds that determine how efficiently MAINVIEW AutoOPERATOR EXECs execute on your system
- AAOTSPxx members contain parameters that control how TapeSHARE operates

From this display, you can

- use the line command **Select** in the **Cmd** column to select a member to view and modify the parameters if you choose
- use the line command **Activate** when you want to dynamically change the BBPARAM member that your system is currently using to manage EXEC thresholds, ALERTs storage, or the Automation Reporter or TapeSHARE applications

The **Activate** command displays the **Confirm Parameter Activation** panel; see Figure 4-13 on page 4-39. This function allows you to verify that you want to dynamically change and use a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOTSPxx member.

For more information about activating your changes, refer to “Dynamically Implementing New Parameter Settings with the **Activate** Command” on page 4-38.

- use the primary command **ADD** to add a new AAOALSxx, AAOARPxx, AAOEXPxx, or AAOTSPxx member

To add a new member or members, type one of the following commands:

**ADD AAOALSxx**

**ADD AAOARPxx**

**ADD AAOEXPxx**

**ADD AAOTSPxx**

on the COMMAND line, where *xx* is a new suffix for a new member. For example, if you want to add a new member named AAOTSP02, type

**ADD AAOTSP02**

## Customizing TapeSHARE

Use the TapeSHARE Parameters panel to see and modify parameters for the TapeSHARE application and control how TapeSHARE automates the sharing of tape devices among your OS/390 images.

Figure 4-12 shows an example of this panel.

**Figure 4-12 TapeSHARE Parameters Panel**

```

BMC Software ----- TapeSHARE Parameters ----- AutoOPERATOR
COMMAND ==>>>                                     TGT --- DSHD
                                                    DATE --- 01/0201//19
                                                    TIME --- 16:27:51

                Member      Current

PARTNER  ==>>>                                * LIST * (Enter ? for a partner list)

TIMEOUT  ==>>>      120          120          Subsystem response timeout value
RETRYCNT ==>>>         2           2           Times to retry a take request
RETRYINT ==>>>        30          30          Interval between take retries
ACTION   ==>>>  DEFAULT      DEFAULT      Allocation failure action

NOGIVE   ==>>>
NOTAKE   ==>>>                                (Enter ? for a device list)
                                                (Enter ? for a device list)

PREF     ==>>>  NO             NO             Enable / Force preferencing
PREFDEV  ==>>>
                                                (Enter ? for a device list)

FREE     ==>>>  DEALLOC      DEALLOC      Deallocation procedure name
TRACE    ==>>>  NO             NO             Start / Stop TapeSHARE tracing

Press END to save changes, CANCEL to cancel changes
    
```

This panel provides two columns of data:

- on the left, the value of the parameters in the AAOTSPxx member for the target BBI-SS PAS

You can enter your modifications on this side of the panel.

- on the right, the value of the parameters that are currently in effect for the target BBI-SS PAS

These values cannot be edited.

The first time you bring up this panel, the display-only fields match the input fields and default values are shown in both columns.

## Describing the Fields

A description of the fields on this panel follows.

<b>Field Name</b>	<b>Description</b>
PARTNER	<p>Specify a BBI-SS PAS name (that can be associated with either one or a range of tape devices) that is a TapeSHARE PLEX partner.</p> <p>You also can include with the BBI-SS PAS name, the device address (or a range of device addresses) that TapeSHARE cannot GIVE to the named partner. See examples below.</p> <p><b>Example 1:</b></p> <p><b>PARTNER=SYSA</b></p> <p>In this example, SYSA is a partner to this image.</p> <p><b>Example 2:</b></p> <p><b>PARTNER=SYSC,0123</b></p> <p>In this example, SYSC is a partner to this image and the device whose address is 0123 cannot be GIVEN to partner SYSC.</p> <p><b>Example 3:</b></p> <p><b>PARTNER=SYSB,0120-0127</b></p> <p>In this example, SYSB is a partner to this image and the devices whose addresses fall between 0120 and 0127 cannot be GIVEN to partner SYSB.</p> <p>If no values are associated with this parameter, TapeSHARE assumes that any BBI node defined in BBINOD00 member is a partner within a TapeSHARE PLEX.</p>
TIMEOUT	<p>Specify the amount of time (in seconds) to wait for the partners to respond to a request for devices.</p>

When this time is reached and a successful GIVE has not been completed, an Allocation Failed Event occurs unless you have specified a number of retries on the RETRYCNT parameter. See below.

**Example:**

**TIMEOUT=20**

Valid values are 0 - 999 seconds and 120 seconds is the default.

**RETRYCNT**

Specify the number of times that TapeSHARE attempts to satisfy a request for devices after an initial attempt has failed.

For example, suppose this system's request for devices is not satisfied and three retries are specified. TapeSHARE attempts three more times to satisfy this request for devices.

**Example:**

**RETRYCNT=3**

This parameter is used in conjunction with the RETRYINT parameter that specifies how long TapeSHARE waits between retry attempts.

Valid values are 0–10 retries and 2 is the default.

**RETRYINT**

Specify the number of seconds that TapeSHARE waits for devices between attempts after an initial attempt has failed.

For example, suppose

- the RETRYCNT= parameter is set to 3 retries
- the RETRYINT= parameter is set to 45 seconds
- this system's request for devices is not satisfied

TapeSHARE waits 45 seconds after the initial attempt fails and retries to satisfy the request (retry attempt 1). If this attempt fails, TapeSHARE waits another 45 seconds and retries (attempt 2). If this attempt also fails, TapeSHARE waits another 45 seconds and retries (attempt 3).

Should all three attempts fail, an Allocation Failed Event occurs and the specified action is taken.

**Example:**

**RETRYCNT=3**

Valid values are 0 - 300 seconds and 30 is the default.

**ACTION** Specify the action that TapeSHARE should take when an Allocation Failed Event occurs because a successful GIVE cannot be performed.

Possible values and their definitions are as follows:

**DEFAULT** specifies that the installation's default action is to occur

**WTOR** issues a WTOR to the operator

If the site does not have a defined default action, this action is the default.

**CANCEL** cancels the job

**NOHOLD** specifies that the job waits without holding resources

**HOLD** specifies that the job waits while holding resources

**Example:**

**ACTION=WTOR**

**NOGIVE** specify a list of device addresses (or a range of addresses) that should not be GIVEN to other request devices to TAKE

**Note:** This action does not prevent you from manually GIVING a NOGIVE device.

If a device address is not associated with this parameter, TapeSHARE assumes that no devices are dedicated to this image.

**Example:**

**NOGIVE=0123**

or

**NOGIVE=0120-012F**

You also can use this parameter in conjunction with the NOTAKE parameter. By specifying a device address on both these parameters, the device will be a NOGIVE-NOTAKE device and it will be excluded completely from TapeSHARE control.

**Example:**

**NOGIVE=0123**

**NOTAKE=0123**

If the device address 0123 is specified on both the NOGIVE and NOTAKE parameters, the 0123 is completely excluded from TapeSHARE control. TapeSHARE does not attempt to GIVE or TAKE this device.

The default value is none.

**NOTAKE**

Specify a list of device addresses (or a range of addresses) that this image should not TAKE when a request for resources is made.

TapeSHARE on this image will not accept (TAKE) the devices specified. If a value is not associated with this parameter, TapeSHARE assumes that it can accept any tape devices when a request for resources is made.

**Example:**

**NOTAKE=0123**

or

**NOTAKE=0120-012F**

You also can use this parameter in conjunction with the NOGIVE parameter. By specifying a device address on both these parameters, the device will be a NOGIVE-NOTAKE device and it will be excluded completely from TapeSHARE control.

For example:

**NOGIVE=0123**

**NOTAKE=0123**

If the device address 0123 is specified on both the NOGIVE and NOTAKE parameters, the 0123 device is completely excluded from TapeSHARE control. TapeSHARE does not attempt to GIVE or TAKE this device.

The default value is none.

**PREF**

Specify whether Preferencing is activated.

If activated, TapeSHARE uses the devices identified on the PREFDEV parameter for only the operation (input or output) identified for that device.

This parameter must be used in conjunction with the PREFDEV parameter.

Specifying YES turns on Preferencing in Guideline mode. For example, in Guideline mode, if SYSA needs an output-only device but only input-only devices are available, the input-only device is GIVEN to SYSA.

Specifying FORCE turns on Preferencing in Force mode. If Force mode were in place, the input-only device would not be GIVEN to SYSA and an Allocation Failure Event would occur.

**Example:**

**PREF=YES**

or

**PREF=NO**

or

**PREF=FORCE**

**PREFDEV**

Specify a tape device address (or a range of tape device addresses) followed by either an **I** for input-only or an **O** for output-only; see examples.

This parameter must be used in conjunction with the PREF parameter.

With the PREF activated, when a device is needed for an output-only operation, TapeSHARE attempts to choose from those devices identified by this parameter as output-only.

If a value is not associated with this parameter, TapeSHARE will not invoke device preferencing.

**Example:**

**PREFDEV=01A0-01A3,I**

or

**PREFDEV=0130,O**

or

**PREFDEV=0150-015F,O**

Default value is none.

FREE

For use with MVS SP4.x only

Specify a 1- to 8-character deallocation procedure which starts after TapeSHARE issues the VARY OFFLINE command to GIVE a device to another image.

Scheduling this procedure triggers deallocation processing in the system GIVING the device and places the device in an OFFLINE state.

The default deallocation name is DEALLOC.

TRACE

Specify that diagnostic information is written to the BBITST DD statement.

**Example:**

**TRACE=NO**

If YES is specified, the BBITST DD statement must appear in the BBI-SS PAS startup JCL for information to be written to it (see BBSAMP member SSJCL).

## Dynamically Implementing New Parameter Settings with the Activate Command

The Dynamic Parameter Manager Activate line command can be used to dynamically change the BBPARM members currently in use by a BBI-SS PAS *without restarting the BBI-SS PAS*.

The command is specified on the main Dynamic Parameter Manager panel (see examples in Figure 4-2 on page 4-4 and Figure 4-3 on page 4-7.)

Figure 4-13 on page 4-39 shows the Confirm Parameter Activation panel that is displayed after the Activate line command is entered.

**Figure 4-13 Confirm Parameter Activation Panel**

```

BMC Software ----- Confirm Parameter Activation ----- AutoOPERATOR
COMMAND ==>                                         TGT --- BBSYSA
                                                    DATE --- 01/02/19
                                                    TIME --- 11:43:13

Current TARGET:      SYSB

Current SSID:        AO22

Last Parm:           AAOEXP00 (Values may have been temporarily modified)

Replaced by:         AAOEXP00
Last update:         01/05/19 11:42
User:                MXY1

Instructions:

  Press ENTER to confirm activation request.
  (The parameters in the current SS will be dynamically updated)

  Press END   to cancel activation request.

```

Automation is suspended briefly while the BBPARM member is read and new control blocks are built. Other products operating in the target BBI-SS PAS continue to operate.

### Describing the Fields

Descriptions of the fields on this panel follow:

Field Name	Description
Current Target	Is the ID of the current target.
Current SSID	Is the ID of the current BBI-SS PAS.
Current Parm	Is the name of the member of BBPARM that was used to build the current parameters in the BBI-SS PAS.
Replaced by	Is the name of the BBPARM member you want to use to build the new set of parameters in the BBI-SS PAS.
Last update	Is the date and time of the last change to this member.
User	Is the user ID that made the last change to this member.



---

# Chapter 5 MVS Console Considerations

MVS commands can be issued by MAINVIEW AutoOPERATOR terminal session (TS) users, EXECs, or Rules. In some cases, the issuer of a command requires a response. MAINVIEW AutoOPERATOR uses MVS consoles to perform this function.

The use of consoles is not a consideration for MVS commands issued from Rules because Rules never expect a response. However, some commands issued by TS users and EXECs require a response to the command and, therefore, require a MVS console.

For more information about how TS users use consoles, refer to “Issuing Commands from Terminal Sessions” on page A-1. For more information about how EXECs use consoles, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* where the IMFEXEC CMD statement is documented.

This chapter describes how to allocate MVS consoles to MAINVIEW AutoOPERATOR subsystems which are used by TS users and EXECs.

The OS/390 consoles used by MAINVIEW AutoOPERATOR are Extended MCS (X-MCS) consoles

## Determining How Many Consoles to Allocate

This section describes how to determine how many consoles MAINVIEW AutoOPERATOR needs to function effectively.

The number of consoles allocated to MAINVIEW AutoOPERATOR controls the number of EXECs that can issue MVS commands and receive a response concurrently. A console is allocated exclusively to an EXEC while a command requesting a response is being processed.

If all of the MAINVIEW AutoOPERATOR consoles are in use, a new command (EXEC) must wait until a console becomes available. This delay could adversely impact the performance of the requesting EXEC and, possibly, the entire MAINVIEW AutoOPERATOR subsystem.

For this reason, BMC Software recommends allocating one console for each EXEC thread defined in MAINVIEW AutoOPERATOR. Refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for information about EXEC threads.

**Note:** The Continuous State Manager (CSM) application requires a minimum of three consoles.

## Extended MCS (X-MCS) Consoles

Unlike subsystem consoles, MVS definitions are not required for X-MCS consoles. MAINVIEW AutoOPERATOR (and other products or subsystems such as NetView) dynamically request that MVS create X-MCS consoles as they are needed. Any number of X-MCS consoles can be created in a MVS system or sysplex.

Prior to MVS Version 4, all consoles have only a 1-byte console ID. Beginning with MVS Version 4, all consoles (subsystem, MCS, and X-MCS) have a 4-byte console ID and an 8-byte console name. MAINVIEW AutoOPERATOR creates all X-MCS console names using the format:

*SSIDnnnn*

where

*SSID* Is the BBI-SS PAS identifier name.

*nnnn* Is a number from one to the total number of X-MCS consoles that is created.

**Note:** You must ensure that no other application uses these console names.

Some consoles may have a 1-byte console ID in addition to the new 4-byte console ID. For example, MCS consoles (defined in the CONSOLxx member of SYS1.PARMLIB) continue to have a 1-byte console ID in addition to the 4-byte console ID and 8-byte console name. However, X-MCS consoles usually do not have a 1-byte console ID; they have only a 4-byte console ID.

This means that applications that interface with consoles specified in the CONSOLxx member of SYS1.PARMLIB do not have to be updated to understand 4-byte console IDs.

Applications that will interface with X-MCS consoles need to be updated to understand 4-byte console IDs. You may have some applications *that do not yet understand 4-byte console IDs*. To remain compatible with these applications, MVS allows some X-MCS consoles to have a 1-byte migration ID (MIGID) specified. Therefore, X-MCS consoles that have a MIGID can interface with applications that have not yet been updated.

In addition, within a sysplex, MVS limits the number of X-MCS consoles with MIGIDs. For this reason, MAINVIEW AutoOPERATOR does not request a MIGID for all X-MCS consoles it creates.

Refer to “Allocating X-MCS Consoles without MIGIDs” for information about allocating X-MCS consoles without MIGIDs. Refer to “Allocating X-MCS Consoles with MIGIDs” for information about allocating X-MCS consoles with MIGIDs.

## Allocating X-MCS Consoles without MIGIDs

The number of X-MCS consoles that MAINVIEW AutoOPERATOR creates without a migration ID is specified with the CONSOLES parameter in BBPARM member BBISSP00. Refer to the *MAINVIEW Common Customization Guide* for more information about specifying values for the CONSOLES parameter in BBPARM member BBISSP00.

## Allocating X-MCS Consoles with MIGIDs

The total number of X-MCS consoles with a MIGID that MAINVIEW AutoOPERATOR creates is specified by using the MIGCONS parameter in BBPARM member BBISSP00.

Within a sysplex, MVS imposes a limit of 150 X-MCS consoles with a MIGID. In addition to the considerations in “Determining How Many Consoles to Allocate” on page 5-2, you must consider this maximum limit when deciding how many X-MCS consoles with a MIGID to specify on the MIGCONS parameter. Refer to the *MAINVIEW Common Customization Guide* for more information about specifying MIGIDs with the MIGCONS parameter in BBPARM member BBISSP00.

MAINVIEW AutoOPERATOR always allocates one extra X-MCS console with a MIGID to support terminal-session users. Refer to “Issuing Commands from Terminal Sessions” on page A-1 for information about TS users who are using consoles with MIGIDs to issue commands.

## Defining X-MCS Consoles to CICS

CICS will not interact with a console unless the console is defined by using CICS Resource Definition. For CICS/ESA version 4.1 and CICS Transaction Server, refer to “CICS/ESA Targets” .

### CICS/ESA Targets

CICS/ESA version 4.1.0 (and later) you need to define all MAINVIEW AutoOPERATOR X-MCS console names. The console names use the format:

*SSIDnnnn*

where

*SSID*            Is the BBI-SS PAS identifier name.

*nnnn*            Is a number from one to the total number of X-MCS consoles created.

---

# Chapter 6 Implementing TapeSHARE

This chapter describes how to implement the TapeSHARE for MAINVIEW AutoOPERATOR component. Some steps are optional.

## Overview

TapeSHARE runs in a BBI-SS PAS and an image can have only one BBI-SS PAS where TapeSHARE is installed. Steps 1 and 2 are required and *they must be performed on every BBI-SS PAS within the TapeSHARE PLEX.*

**Note:** To use TapeSHARE, you must make sure the OS/390 Allocated/Offline Device Exit is not specified on any image where TapeSHARE will run. TapeSHARE is defined dynamically as an Allocated/Offline Device Exit to OS/390 when it initializes.

## Prerequisites

To use TapeSHARE, the only prerequisites are that you have a valid MAINVIEW AutoOPERATOR for OS/390 password key and that you have a password key for the TapeSHARE for MAINVIEW AutoOPERATOR component. See “Step 1: Entering the Password Key for TapeSHARE (Required)” for more information about password keys.

## Step 1: Entering the Password Key for TapeSHARE (Required)

1. TapeSHARE requires that you have a password key to use it.

Once you have a key, you must enter it in the BBPARM member BBKEYS. Refer to the section “Specify Product Option Password Keys” in the *MAINVIEW Common Customization Guide* for information about acquiring and using a password key for TapeSHARE.

2. Go to “Step 2: Activating TapeSHARE (Required)”.
3. Repeat these steps for every BBI-SS PAS that is a partner in the TapeSHARE PLEX as part of the TapeSHARE PLEX installation.

## Step 2: Activating TapeSHARE (Required)

After the password key is entered into BBPARM member BBKEYS, all you need to do is restart the BBI-SS PAS and TapeSHARE is activated for that image. *You do not have to restart OS/390.*

For every image that you want to include in a TapeSHARE PLEX, you must add a password key to the BBPARM member BBKEYS for that image and restart the BBI-SS PAS as described in “Step 1: Entering the Password Key for TapeSHARE (Required)”.

### Using TapeSHARE in its Default State

TapeSHARE is designed for you to install and use in its out-of-box default state. If you choose to use TapeSHARE in its default state, every BBI node defined in the BBINOD00 member with a TapeSHARE installed will be considered a partner in the TapeSHARE PLEX.

For more information about what values are used in the default state, refer to “Modifying TapeSHARE for MAINVIEW AutoOPERATOR Parameters in AAOTSPxx” on page 4-30.

## Step 3: Securing TapeSHARE (Optional)

Complete this optional step if you decide to secure who has access to the TapeSHARE for MAINVIEW AutoOPERATOR component.

Security for all MAINVIEW AutoOPERATOR options is described in the BMC Software document *Implementing Security for MAINVIEW Products*.

Securing access to TapeSHARE is the same as securing access to any of the MAINVIEW AutoOPERATOR applications, which is described in “Chapter 4, Reviewing Product Resources” of the *Implementing Security for MAINVIEW Products* manual.

The resource name for securing access to the TapeSHARE component is *prefix.ssid.AAO.target.TAPSHR*.

## **After TapeSHARE Is Installed**

After TapeSHARE is installed, it can perform all of your tape device sharing without ever performing any customization steps. Refer to the *MAINVIEW AutoOPERATOR Options User Guide* for information about how to customize and use TapeSHARE.



---

---

# Chapter 7 Implementing Automatic Restart Manager (ARM)

This chapter describes implementation considerations for the Automatic Restart Manager (ARM) and how to implement support for it.

**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 later) and MAINVIEW AutoOPERATOR 4.1 (or later).

## Overview

The Automatic Restart Manager (ARM) is an IBM facility that allows MVS/ESA to automatically restart Started Tasks (excluding TSO users) after abnormal termination.

ARM support is available in either single or multisystem sysplex using couple data sets. By implementing support for ARM, you can ensure that MAINVIEW AutoOPERATOR will be restarted quickly and automatically after an abend with no operator intervention required.

Previously, to ensure MAINVIEW AutoOPERATOR availability, you had to use a second MAINVIEW AutoOPERATOR to monitor the primary MAINVIEW AutoOPERATOR subsystem or use an outboard processing facility such as Elan. If the primary MAINVIEW AutoOPERATOR abended, user-defined automation routines in the secondary MAINVIEW AutoOPERATOR (or Elan) would attempt to restart MAINVIEW AutoOPERATOR.

If your system is using MVS/ESA 5.2 or later, you can choose to implement ARM support for the MAINVIEW AutoOPERATOR. This eliminates the need for a secondary MAINVIEW AutoOPERATOR system (or Elan) to monitor MAINVIEW AutoOPERATOR availability and can simplify MAINVIEW AutoOPERATOR restarts by allowing MVS/ESA to monitor and restart MAINVIEW AutoOPERATOR.

## Implementation Considerations

To implement ARM support for MAINVIEW AutoOPERATOR, you must modify the ARMPOLCY parameter in BBPARM member BBISSP00 (see “Enabling ARM Support” on page 7-3). The following choices are available:

- Do not implement ARM support.
- Implement ARM support with the default ARM policy.
- Implement ARM support with your own user-designed policy.

### Implementing ARM Support: Default ARM Policy

The default ARM restart policy is that ARM attempts to restart an abended Started Task up to three times in a five-minute interval. After three attempts and the Started Task has still failed to restart, ARM stops attempting restarts and a message is written to SYSLOG.

Therefore, if you choose to use the default ARM policy for MAINVIEW AutoOPERATOR, ARM will attempt to restart an abended MAINVIEW AutoOPERATOR subsystem up to three times in a five-minute interval.

The default ARM policy uses the same start command that was used to start the MAINVIEW AutoOPERATOR subsystem originally. After the third attempt, if MAINVIEW AutoOPERATOR has not successfully restarted, a message is written to SYSLOG and restart attempts stop.

If you decide that ARM should perform more restarts or perform them over a different period of time, you can write your own ARM restart policy. Refer to “Implementing ARM Support: Creating a User-Defined Policy”.

## Implementing ARM Support: Creating a User-Defined Policy

You might already have an ARM restart policy in place or you can create your own ARM restart policy. To create your own policy, refer to the IBM publication *Setting Up a Sysplex*.

In either case, MAINVIEW AutoOPERATOR provides support for a user-defined ARM restart policy. “Enabling ARM Support” describes how to enter the name of a user-defined ARM restart policy in BBPARM member BBISSP00.

**Note:** Only one ARM policy can operate in an OS/390 image at one time.

## Enabling ARM Support

To enable ARM support, edit the BBPARM member BBISSP00 and specify

**ARMPOLCY=[NO|YES|xxxxxxx]**

where valid values are

### **NO**

Specifies that the BBI-SS PAS will not use the ARM facility; the default setting

### **YES**

Specifies that the BBI-SS PAS will use the ARM default restart policy

The default ARM restart policy is that ARM will restart an abended address space up to three times within five 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

Once you have completed modifying the ARMPOLCY parameter, restart the BBI-SS PAS. Either a warm start or a cold start will cause the new ARM policy to be recognized.

AutoCustomization provides a step for ARM support implementation. The *MAINVIEW Common Customization Guide* describes how to implement ARM support during manual customization.

## Stopping MAINVIEW AutoOPERATOR When ARM Is Enabled

You might find that you need to stop MAINVIEW AutoOPERATOR when ARM is enabled and you do not want ARM to restart MAINVIEW AutoOPERATOR. To stop (or cancel) MAINVIEW AutoOPERATOR without ARM attempting a restart:

- Use the **STOP|P** command for a normal shutdown of MAINVIEW AutoOPERATOR; for example:

```
P ao_ssid
```

- Use the **CANCEL|C** or **FORCE** command if you unexpectedly need to cancel MAINVIEW AutoOPERATOR; for example:

```
C ao_ssid
```

or

```
FORCE ao_ssid
```

where `ao_ssid` is the subsystem ID of the MAINVIEW AutoOPERATOR that you want to stop or cancel.

---

# Chapter 8 Implementing Support for MPF

This chapter describes how MAINVIEW AutoOPERATOR can coexist with the MVS Message Processing Facility (MPF). If your site does not use MPF, you can skip this chapter.

## Overview

The MVS Message Processing Facility (MPF) allows you to modify the attributes of write-to-operator messages (WTOs). For example, with MPF you can modify the appearance of a WTO (such as its color and highlighting) and suppress messages from consoles.

Previous releases of MAINVIEW AutoOPERATOR provided the HONORMPF=(YES | NO) parameter, which determined how MAINVIEW AutoOPERATOR automated WTOs with the MPF SUP keyword setting.

If HONORMPF is set to the default value of NO:

```
HONORMPF=NO
```

all WTOs are eligible for MAINVIEW AutoOPERATOR automation (regardless of how the SUP keyword is set on the WTO).

If HONORMPF is set to YES:

```
HONORMPF=YES
```

only WTOs with the SUP keyword set to NO are eligible for MAINVIEW AutoOPERATOR automation.

In MAINVIEW AutoOPERATOR 4.1, you can specify that MAINVIEW AutoOPERATOR will determine which WTOs are eligible for automation by checking the WTO for both the SUP and the AUTO keyword setting.

The possible values of the MPF AUTO keyword are

AUTO= ( YES | NO | token )

where `token` is an up to 8-character user-specified token name. The token name can contain wildcards and blanks.

The possible values of the MPF SUP keyword are

SUP= ( YES | NO )

**Note:** For more information about MPF, refer to the IBM publication *MVS/ESA Initialization and Tuning Reference*.

To check a WTO for its AUTO and SUP settings, use the MPFFILTR= parameter in BBPARM member AAOPRM00. For more information, refer to “Implementation Considerations”.

## Implementation Considerations

Use the MPFFILTR parameter when you want MAINVIEW AutoOPERATOR to check the MPF AUTO and SUP settings of a WTO to determine if the WTO should be eligible for MAINVIEW AutoOPERATOR automation.

**Note:** You cannot use the MPFFILTR parameter if HONORMPF=YES is set. If HONORMPF=YES is set, MAINVIEW AutoOPERATOR ignores any settings on the MPFFILTR= parameter.

When the MPFFILTR parameter is set, MAINVIEW AutoOPERATOR actively checks the AUTO and SUP settings and only WTOs that match the MPFFILTR criteria are eligible for MAINVIEW AutoOPERATOR automation.

For more information, refer to “Setting the MPFFILTR Parameter” on page 8-3.

## Setting the MPFFILTR Parameter

The MPFFILTR parameter allows you specify how MAINVIEW AutoOPERATOR handles WTOs that have the MPF AUTO and SUP keywords associated with them. The syntax for MPFFILTR parameter is

```
MPFFILTR=( AUTO=YES | NO | token , SUP=YES | NO )
```

The AUTO and SUP keywords always must be used together on the MPFFILTR statement. You cannot use one keyword without the other.

The possible combinations of the AUTO and SUP keywords are

```
MPFFILTR=      ( AUTO=NO , SUP=YES )
                ( AUTO=NO , SUP=NO )
                ( AUTO=YES , SUP=YES )
                ( AUTO=YES , SUP=NO )
                ( AUTO=* , SUP=NO )
                ( AUTO=* , SUP=YES )
                ( AUTO=NO , SUP=* )
                ( AUTO=YES , SUP=* )
                ( AUTO=TOKEN , SUP=YES )
                ( AUTO=TOKEN , SUP=NO )
                ( AUTO=TOKEN , SUP=* )
```

You can select up to four of the possible combinations. Once you use any of the combinations, MAINVIEW AutoOPERATOR checks all WTOs for a match between the WTO's settings and the MPFFILTR setting.

If the WTO's setting matches the MPFFILTR setting, the WTO is eligible for automation by MAINVIEW AutoOPERATOR. The following examples describe some of the possible MPFFILTR settings.

### Example

If the MPFFILTR parameter is set

```
MPFFILTR=( AUTO=NO , SUP=NO )
```

only WTOs that have both AUTO and SUP set to NO are eligible for MAINVIEW AutoOPERATOR automation. MAINVIEW AutoOPERATOR ignores all other messages that have different AUTO and SUP settings; they are not eligible for MAINVIEW AutoOPERATOR automation.

**Example**

If the MPFFILTR parameter is set

```
MPFFILTR=( AUTO=NO , SUP=YES )
```

only WTOs that have these AUTO and SUP settings are eligible for automation by MAINVIEW AutoOPERATOR. MAINVIEW AutoOPERATOR ignores all other messages that have different AUTO and SUP settings; they are not eligible for MAINVIEW AutoOPERATOR automation.

**Example**

If the MPFFILTR parameter is set

```
MPFFILTR=( AUTO=BOOLE , SUP=NO )
```

only WTOs with an AUTO setting of BOOLE and a SUP setting of NO are eligible for automation by MAINVIEW AutoOPERATOR.

If you want all WTOs that have a MPF token setting of BOOLE and you do not want to check the SUP setting, use

```
mpffiltr=( AUTO=BOOLE , SUP=* )
```

All WTOs with an AUTO setting of BOOLE are eligible for automation by MAINVIEW AutoOPERATOR, regardless of its SUP setting. So, for example, if a WTO has any of the following settings, they are not eligible for automation:

```
AUTO=YES , SUP=NO  
AUTO=NO , SUP=NO  
AUTO=YES , SUP=YES  
AUTO=NO , SUP=NO  
AUTO=BO OL , SUP=NO  
AUTO=NETVIEW , SUP=YES
```

**Example**

To use more than one set of AUTO and SUP settings

```
MPFFILTR=( ( AUTO=YES , SUP=YES ) , ( AUTO=YES , SUP=NO ) )
```

Now all WTOs with an auto setting of YES are eligible for automation regardless of the setting. You get the same result when you set MPFFILTR to

```
mpffiltr=(AUTO=YES,SUP=*)
```

MAINVIEW AutoOPERATOR ignores all other WTOs that have different AUTO and SUP settings; they are not eligible for MAINVIEW AutoOPERATOR automation.



---

## Chapter 9    **Activating the REXX/370 Alternate Library**

This chapter describes how to implement the REXX/370 Alternate Library.

If your site does not have the IBM library for SAA REXX/370 installed and you want to run compiled REXX programs in a BBI-SS PAS, you must follow the instructions in this chapter to implement the REXX/370 Alternate Library.

If the IBM library for SAA REXX/370 is available, no customization is required on the MAINVIEW AutoOPERATOR side. Please refer to the IBM documentation for any steps required to make the library accessible to the BBI-SS PAS.

Make sure that the BBI-PAS has a STEPLIB to the BBLINK data set containing the REXX/370 Alternate Library modules. If the BBI-PAS does *not* have a STEPLIB (for example, you place the BBLINK data set in the LINKLST), the BBI-PAS loads IRXCMPTM from the LPA instead of the BBLINK data set (as a result of the normal MVS search order). You must load IRXCMPTM from the BBLINK data set in order to activate the REXX/370 Alternate Library.

To activate the REXX/370 Alternate Library:

**Step 1** Review the sample job distributed in BBSAMP member BBREXALT.

This job activates the REXX/370 Alternate Library. The REXX/370 Alternate Library must be activated at sites that need to execute compiled REXX programs in a BBI-SS PAS *but do not have the IBM library for SAA REXX/370 (program number 5695-014) available.*

**Note:** This job should be not be used at sites where the IBM library for SAA REXX/370 is available.

**Step 2** Edit the job, if necessary.

**Step 3** Follow all customization instructions.

**Step 4** Submit the job.

The REXX/370 Alternate Library is installed.

## SMP/E Messages

You might receive the following SMP/E messages when you install compiled REXX programs. These are benign messages and you can ignore them.

### For SMP/E release 1.4, you might receive

```
GIM4441 ASSEMBLY FOR SRC aaaaaaa in SYSMOD BPOnnnn  
      WILL NOT BE DONE - RELATED MOD ENTRY NOT FOUND
```

### For SMP/E release 1.5 and later, you might receive

```
GIM44402W aaaaaaa WAS NOT ASSEMBLED FOR SRC aaaaaaa  
      IN SYSMOD BPOnnnn BECAUSE THERE IS NO MOD  
      ENTRY FOR aaaaaaa. SMP/E CANNOT DETERMINE  
      THE TARGET LIBRARY FOR THE ASSEMBLER OUTPUT.
```

These messages are issued because compiled REXX elements are distributed as ++SRC elements.

---

---

# Chapter 10 Defining OSPI Virtual Terminals

This chapter describes defining OSPI virtual terminals.

## Overview

MAINVIEW AutoOPERATOR provides the Open Systems Procedural Interface (OSPI) as an interface to VTAM-based products. OSPI provides a means for REXX- or CLIST-based automation procedures to interface with any LU2 VTAM application that uses full screens to communicate with users.

With OSPI, MAINVIEW AutoOPERATOR has logon capabilities and complete access to any VTAM application's data stream. In this way, MAINVIEW AutoOPERATOR can interact with the application by analyzing the output data and issuing the VTAM application's own commands.

By automatically interfacing with critical VTAM applications and simulating a user at a VTAM terminal, OSPI can communicate with various data center software products and decrease the number of VTAM-based terminals.

This terminal emulation requires that some number of OSPI virtual terminals be defined to VTAM. In addition, some applications, such as CICS and IMS, might require local definitions for OSPI terminals. Refer to "OSPI Terminal Definitions for CICS and IMS" on page J-1 for more information.

To use OSPI, you must be running VTAM 3.0 or later.

- "Defining the ACB Pool" on page 10-3 contains instructions for defining OSPI virtual terminals to VTAM.

- “Defining the ACB Pool to MAINVIEW AutoOPERATOR” on page 10-3 contains instructions for defining OSPI virtual terminals to MAINVIEW AutoOPERATOR.
- “OSPI Application Definitions” on page 10-4 contains information about local terminal definitions that might be required for some applications.

## Using the OSPI ACB Pool

When OSPI attempts to establish a session with a VTAM application, a VTAM ACB is required to represent the OSPI virtual terminal. If a specific ACB name is not requested, the OSPI ACB pool is searched for an available and usable ACB.

See the chapters that describe the OSPI Scripting facility and the IMFEXEC LOGON command in the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for information about requesting a specific ACB.

## Naming Conventions

The pool consists of a group of ACBs named using the following conventions:

*PPPPnnnn*

where

*PPPP*            Is the 4-character pool prefix.

*nnnn*            Is a sequential number from 0000 to 9999.

The default pool prefix is OSPI. You can specify a different prefix by using the OSPIPRFX parameter in BBPARM member AAOEXP00.

**Note:** The prefix *must* be four characters.

OSPI attempts to locate the first usable ACB beginning with OSPI0000. The search continues sequentially until a usable ACB is found or all OSPI ACBs have been exhausted.

## Defining the ACB Pool

A VTAM APPL statement is required for each ACB that you want to define in the OSPI ACB pool. You can add new APPL statements to an existing member or create a new member in SYS1.VTAMLST. The statements should be similar to the following example:

```
OSPI      VBUILD TYPE=APPL
OSPI0000 APPL AUTH=(ACQ,NOTSO,VPACE)
OSPI0001 APPL AUTH=(ACQ,NOTSO,VPACE)
```

The number of ACBs defined determines the maximum number of concurrent OSPI sessions MAINVIEW AutoOPERATOR can support unless ACBs outside the OSPI pool are also used.

ACBs must be activated before they can be used. They may be activated automatically at VTAM start-up or by issuing a VTAM VARY command.

If LOGMODE entries other than the entries defined in the default MODETAB will be used by OSPI sessions, add the MODETAB= keyword to each APPL statement.

## Defining the ACB Pool to MAINVIEW AutoOPERATOR

When OSPI attempts to locate a usable ACB in the pool, the search begins with OSPI0000 and continues sequentially until a usable ACB is found or all OSPI ACBs have been exhausted.

If the number of ACBs in the OSPI pool has not been defined to MAINVIEW AutoOPERATOR, OSPI assumes that the highest numbered (last) ACB has been used when an open is unsuccessful against an ACB that is not in use. The search will not continue even if higher numbered ACBs might be usable.

BMC Software recommends that you define the number of ACBs in the pool using the OSPINUM parameter in BBPARM member AAOEXP00. This causes a pool search to continue until all OSPI ACBs have been examined. The number specified using the OSPINUM parameter should match the number of ACBs defined to VTAM.

## OSPI Application Definitions

Some applications, such as CICS and IMS, require local definitions for OSPI terminals. Refer to “OSPI Terminal Definitions for CICS and IMS” on page J-1 for information about how to define virtual terminals to CICS and IMS.

When defining OSPI virtual terminals to applications other than CICS and IMS, BMC Software recommends that the terminals be defined as 3278 model 2.

---

---

# Chapter 11 Implementing BBI-SS PAS to CICS Communication

This chapter describes the steps required to implement BBI-SS PAS to CICS communication.

## Overview

For sites implementing the MAINVIEW for CICS product *at the same time* as MAINVIEW AutoOPERATOR for CICS, use the *MAINVIEW for CICS Customization Guide* and perform the steps described in “Chapter 3: Standard Implementation Procedures”

This chapter discusses implementation procedures if you are installing MAINVIEW AutoOPERATOR for CICS *only*.

Certain MAINVIEW AutoOPERATOR for CICS functions require BBI-SS PAS to CICS communication. However, numerous MAINVIEW AutoOPERATOR for CICS functions are available to you if you do not implement BBI-SS PAS-to-CICS communication. See “MAINVIEW AutoOPERATOR for CICS Functions (with BBI-SS PAS-to-CICS Communication)” on page 11-7 for additional information.

The following steps are required to implement BBI-SS PAS to CICS communication

- Review implementation considerations (page 11-2).
- Specify resources to CICS (page 11-3).
- Modify CICS startup JCL (page 11-5).
- Restart CICS target regions (page 11-6).

To begin implementing BBI-SS PAS-to-CICS communication, go to “Implementation Considerations”.

## Implementation Considerations

This section provides information you should consider when you are implementing MAINVIEW for CICS or MAINVIEW AutoOPERATOR for CICS or both products together.

## Site-Specific Security

Transactions FST2, JNL2, FIC2, FCM1, and FCD2 run asynchronously. The TRANSEC parameter of the transaction definition must have a value of 1 when CICS security is used. If your site uses an external security package, such as RACF or ACF2, that package must be updated to let these transactions run asynchronously.

Transaction SMN2 can be RACF-secured.

## Temporary Storage Table (TST) Compatibility

To avoid conflicts between the BMC Software temporary storage prefix and any existing recoverable temporary storage definitions, perform the following steps for each CICS that will communicate with a BBI-SS PAS:

**Step 1** Determine if a Temporary Storage Table (TST) is now in use.

If a TST is not being used, there is no conflict.

**Step 2** For an existing TST, determine if the DATAID parameter for any of the DFHTST TYPE=RECOVERY or TYPE=REMOTE macros specifies a value that conflicts with the following string:

CMR.I

If there is a conflict, see Appendix B, “Changing the Use of CICS Recoverable Temporary Storage”.

# Specifying Resources to CICS

The following sections describe the steps necessary to make the MAINVIEW AutoOPERATOR transactions and programs available to CICS.

## Making the MAINVIEW AutoOPERATOR Transactions and Programs Available to CICS/ESA and CICS Transaction Server

The member CMRCSDES in the **prefix.BBSAMP** data set contains JCL that refers to other **prefix.BBSAMP** members that contain in-line PCT and PPT definitions.

Use this JCL to define BMC Software transactions and programs required for BBI-SS PAS to CICS communication by doing the following steps:

- Step 1** Copy CMRCSDES from **prefix.BBSAMP** to a data set that is not a BMC Software product target or distribution library. Make sure the new member name conforms to site naming conventions.
- Step 2** Edit the member created in Step 1. Read the comments in the member and modify the JCL accordingly.
- Step 3** Submit the JCL and check the output. Be sure the return code is not higher than 4.
- Step 4** Verify that the BMC Software transaction IDs do not conflict with any existing transaction IDs.

If a conflict exists, and the existing transaction IDs cannot be changed, see Appendix C, “CICS Transaction and Program Names”.

The MAINVIEW AutoOPERATOR transactions and programs are now available to CICS.

Go to “Modifying the Program List Table (PLT)” .

## Modifying the Program List Table (PLT)

The PLT defines programs to be executed in the post-initialization phase of CICS startup.

Completing the steps in this section automates the initiation of BBI-SS PAS-to-CICS communication at CICS startup.

Use of the PLT is optional. However, if you do not use the PLT, manually start BBI-SS PAS-to -CICS communication using either the SMN2 transaction from within the CICS region or the FST2 transaction from the OS/390 console.

Refer to Appendix D, “Manual Activation of BBI-SS to CICS Communication” for more information.

<b>Assembly Note</b>
To assemble the table changes, concatenate BBSAMP with SYSLIB in your site's CICS table assembly JCL (normally, DFHAUPLE).

**Existing PLT**

**Step 1** Edit the PLT and find the following statement at the end of the table:

```
DFHPLT TYPE=FINAL
```

**Step 2** Just before the DFHPLT TYPE=FINAL statement, insert the following macro invocation:

```
CMRPLT RELEASE=41 | 12 | 13 | 21 | 22
```

The parameters for the CMRPLT macro are described in Table 11-1.

**Table 11-1 Syntax for the CMRPLT Macro**

Term	Parameters	Description
RELEASE	41 12 13 21 22	Specifies the release of the CICS system that will use the resulting table (for example, CICS 4.1.0, 2.1.0, 2.2.0).
SUFFIX	BI	Specifies the suffix for the new PLT. Specify a suffix that adheres to the naming conventions at your site. BI is the default. If you do not use the default, create an entry for DFHPLTxx (where xx is the suffix) in the PPT table of the CICS system that uses the PLT. When SUFFIX is used, TYPE=INITIAL and TYPE=FINAL statements are generated automatically.

**Step 3** Assemble the newly revised PLT.

The PLT is modified.

## No Existing PLT

- Step 1** Create a new PLT in your CICS table library using a member name that adheres to the naming conventions at your site. The new member should contain the following macro invocation:

```
CMRPLT RELEASE=41 | 12 | 13 | 21 | 22
```

The parameters for this macro are described in Table 11-1 on page 11-4.

- Step 2** Assemble the newly created PLT.
- Step 3** Add the following parameter to the CICS initialization override parameters:

```
PLTPI=xx
```

where *xx* is the suffix specified in Step 1 above.

The PLT is created and enabled.

Go to “Modifying CICS Startup JCL” .

## Modifying CICS Startup JCL

Do the following steps for each CICS that will communicate with a BBI-SS PAS:

- Step 1** Add the following DD statement to the CICS execution step:

```
//BBIPARM DD DSN=prefix.UBBPARM,DISP=SHR
```

- Step 2** Add the following DD statement to the DFHRPL library concatenation in the CICS execution step:

```
// DD DSN=prefix.BBLINK,DISP=SHR
```

The CICS startup JCL is modified.

## Restarting CICS Target Regions

To complete the implementation of BBI-SS PAS to CICS communication, do the following steps:

- Step 1** Start each BBI-SS PAS that will communicate with a CICS region (if it is not already active).
- Step 2** Shut down each CICS region that will communicate with a BBI-SS PAS that is already active.
- Step 3** Start each CICS region that will communicate with a BBI-SS PAS.
- Step 4** For each CICS region that communicates with a BBI-SS PAS but for which the procedure in “Modifying the Program List Table (PLT)” on page 11-3 was not performed, manually start BBI-SS PAS-to-CICS communication; use either the SMN2 transaction from within the CICS region or the FST2 transaction from the OS/390 console.

Refer to “Manual Activation of BBI-SS to CICS Communication” on page D-1 for more information.

- Step 5** Ensure that the following messages are issued from each CICS region that will communicate with a BBI-SS PAS:

```
+FT431I SESSION WITH BBI-SS PAS ESTABLISHED  
+FT428I CICS MESSAGE INTERCEPT COMPLETED
```

The implementation of BBI-SS PAS to CICS communication is now complete.

# MAINVIEW AutoOPERATOR for CICS Functions (with BBI-SS PAS-to-CICS Communication)

If you choose to implement BBI-SS PAS-to-CICS communication, you will be able to

- use Rules to invoke CICS-dependent commands without using an EXEC
- use the Rule Processor to capture messages from the CICS transient data queue
- use hyperlinks to invoke MAINVIEW for CICS applications
- use the BROADCAST application from the CICS Operator Workstation in the BBI-TS
- use the CICS-dependent IMFEXEC CICS commands from an EXEC

The CICS-dependent IMFEXEC CICS commands are as follows:

<b>ACQUIRE TERMINAL</b>	Acquire a VTAM-supported terminal.
<b>ALLOC</b>	Allocate a data set to a CICS region.
<b>CEMT</b>	Issue a CICS master terminal command.
<b>CHAP</b>	Change a task's priority in a CICS region.
<b>CLOSE</b>	Close a file in a CICS region.
<b>DISABLE</b>	Disable a CICS resource.
<b>DROP</b>	Decrease the use count of a CICS program.
<b>DUMPDB</b>	Prepare a database for dumping.
<b>ENABLE</b>	Make a CICS resource available for use.
<b>FREE</b>	Deallocate a data set from a CICS region.
<b>INSERVE</b>	Place a CICS resource in service.
<b>KILL TASK(PURGE, FORCEPURGE)</b>	Terminate a task in a CICS region by task number.
<b>KILL TERM</b>	Terminate a task in a CICS region by term ID.
<b>LOAD</b>	Load a program in a CICS region.
<b>NEWCOPY</b>	Load a new version of a program in a CICS region.
<b>OPEN</b>	Open a file in a CICS region.
<b>OUTSERVE</b>	Place a CICS resource out of service.
<b>PURGE</b>	Purge a resource in a CICS region.
<b>RECOVERYDB</b>	Prepare a database for recovery.

<b>RELEASE TERMINAL</b>	Release a VTAM-supported terminal.
<b>STARTDB</b>	Start a database.
<b>STOPDB</b>	Stop a database.

## MAINVIEW AutoOPERATOR for CICS Functions (without BBI-SS PAS-to-CICS Communication)

If you choose *not* to implement BBI-SS PAS-to-CICS communication, you can still use MAINVIEW AutoOPERATOR for CICS to

- use the STATUS application from the CICS Operator Workstation in the BBI-TS
- use Rules to invoke CICS-independent commands without needing an EXEC
- use the CICS-independent IMFEXEC CICS commands from an EXEC

The CICS-independent IMFEXEC commands are as follows:

<b>ALLOC (LOCAL only)</b>	Allocate a data set to the BBI-SS.
<b>ALTER</b>	Change CICS task-related throttles.
<b>ALTERVS</b>	Change virtual storage in the CICS region.
<b>DISABLE (TRAN, DEST, PROGRAM)</b>	Disable a CICS resource.
<b>ENABLE (TRAN, DEST, PROGRAM)</b>	Make a CICS resource available for use.
<b>FREE (LOCAL only)</b>	Deallocate a data set from the BBI-SS.
<b>KILL TASK (KILL, WITH DUMP, FORCE)</b>	Terminate a task in a CICS region by task number.
<b>QUERY</b>	Invoke a MAINVIEW for CICS service.
<b>SPURGE</b>	Change the spurge value for CICS transactions.



---

---

# Chapter 12 MAINVIEW AutoOPERATOR for CICS Optional Tailoring

This section describes optional steps for tailoring MAINVIEW AutoOPERATOR for CICS to your site's requirements.

## Overview

All steps described in this section are optional, and the full functionality of MAINVIEW AutoOPERATOR for CICS is available whether or not they are performed.

The MAINVIEW AutoOPERATOR for CICS optional tailoring steps are as follows:

- “Customizing Terminal Types Eligible for Broadcast Messages (CAOTTAB)” on page 12-2
- “Limiting CICS Transient Data Queues Eligible for Rule Processing (CAODTAB)” on page 12-3

If customized to do so, MAINVIEW AutoOPERATOR for CICS can intercept any data written to the Intrapartition Transient Destinations as well.

## Customizing Terminal Types Eligible for Broadcast Messages (CAOTTAB)

When the MAINVIEW AutoOPERATOR for CICS BROADCAST application is invoked, a list of all terminals eligible for the BROADCAST function is displayed. The following criteria must be met before a terminal is eligible for broadcasting:

- It must have a CICS status of INSERVICE.
- If it is a VTAM terminal, it must be currently acquired.
- Its model specification must match the specifications of the supplied terminal table.

Because terminal models can differ from installation to installation, you may need to customize the MAINVIEW AutoOPERATOR for CICS terminal table so that it includes the terminal types used at your site.

The CAOTTAB macro is used to generate the MAINVIEW AutoOPERATOR for CICS terminal table. The default table is distributed by BMC Software using the following CAOTTAB macro invocation:

```
CAOTTAB (L3277,R3277)
```

Any terminal type that can be specified for the CICS terminal definition operand TERMMODL can be specified for CAOTTAB.

The member TTABJCL in **prefix.BBSAMP** contains sample JCL that can be used to assemble and link CAOTTAB as follows:

1. Copy TTABJCL from **prefix.BBSAMP** to **prefix.UBBSAMP**. Make sure the new member name conforms to site naming conventions.
2. Edit the member created in Step 1. Read the comments in the member and then modify the JCL accordingly.
3. Modify the sample invocation of the CAOTTAB macro so that all the terminal types required are eligible for BROADCAST messages.
4. Submit the newly created JCL; check the output to insure that all steps complete with a return code of 0.

**Note:** S106 abends can occur if the newly link-edited module causes the BBLINK data set to enter secondary extents and the active BBI-SS PAS attempts to load the module. If this happens, stop BBI-SS PAS and restart it.

## Limiting CICS Transient Data Queues Eligible for Rule Processing (CAODTAB)

By default, MAINVIEW AutoOPERATOR for CICS intercepts any data written to any CICS extra partition transient data destination. If customized to do so, MAINVIEW AutoOPERATOR for CICS also can intercept any data written to intrapartition destinations. It then runs its Rule Processor definitions to determine if a particular message should be intercepted and acted upon.

The default may be acceptable in a CICS system that does not send large amounts of user data to its transient data destinations. But if there is a large volume of data traffic, performance can be impacted.

If your CICS regions have a large volume of transient data traffic and you want to limit the names of the CICS transient data destinations that are eligible for the Rule Processor message interception, use the CAODTAB macro to generate a CICS transient data name table that can define which queue names are part of the table and that can

- include only selected intrapartition transient data queue destinations
- include only selected extrapartition transient data queue destinations
- include both selected extrapartition and intrapartition transient data queue destinations
- exclude only selected intrapartition transient data queue destinations
- exclude only selected extrapartition transient data queue destinations
- exclude both selected extrapartition and intrapartition transient data queue destinations

Refer to “Using the CAODTAB Macro” for more information about how the CAODTAB macro works.

Any CICS extrapartition or intrapartition transient data destination name defined at your site can be specified for CAODTAB.

## Using the CAODTAB Macro

The member DTABJCL in **prefix.BBSAMP** contains sample JCL which you can use to assemble and link CAODTAB:

1. Copy DTABJCL from **prefix.BBSAMP** to **prefix.UBBSAMP**. Make sure the new member name conforms to site naming conventions.
2. Edit the member created in Step 1. Read the comments in the member and then modify the JCL accordingly.
3. Modify the sample invocation of the CAODTAB macro so that it uses applicable parameters from “Describing CAODTAB Macro Parameters”.
4. Submit the newly created JCL; check the output to ensure that all steps are completed with a return code of 0.

## Describing CAODTAB Macro Parameters

The following list describes input parameters for the CAODTAB macro:

<b>Parameter</b>	<b>Possible values</b>
EXTRA	(YES   NO,INCLUDE   EXCLUDE)
INTRA	(YES   NO,INCLUDE   EXCLUDE)
QUEUES	[(name1),(name2),(namen)...] The QUEUES= parameter is optional. Use the QUEUES= parameter when you want to explicitly include or exclude specific queues by queue name.

## Examples

The following examples show how to use different combinations of the parameters on the CAODTAB macro invocation.

### Example

To specify that only extrapartition transient data messages will be eligible for processing by the Rule Processor:

```
EXTRA=( YES , INCLUDE ) , INTRA=( NO ) , QUEUES=( QUE1 , QUE2 , QUE3 )
```

The CICS transient data name table will be an inclusion table for extrapartition transient data messages from queues named QUE1, QUE2, and QUE3.

If no queue names are specified (for example, if QUEUES=( ) or if the queues parameter is not used), all extrapartition transient data messages are included.

### Example

To specify that only intrapartition transient data messages will be eligible for processing by the Rule Processor:

```
EXTRA=( NO ) , INTRA=( YES , INCLUDE ) , QUEUES=( QUE4 , QUE5 , QUE6 )
```

The CICS transient data name table will be an inclusion table for intrapartition transient data messages from queues named QUE4, QUE5, and QUE6.

If no queue names are specified (for example, if QUEUES=( ) or if the queues parameter is not used), all intrapartition transient data messages are included.

**Example**

To specify that both intrapartition and extrapartition transient data messages will be eligible for processing by the Rule Processor:

```
EXTRA=( YES , INCLUDE ) , INTRA=( YES , INCLUDE ) ,  
QUEUES=( QUE7 , QUE8 )
```

The CICS transient data name table will be an inclusion table for both extrapartition and intrapartition transient data messages for queues named QUE7 and QUE8.

If no queue names are specified (for example, if QUEUES=( ) or if the queues parameter is not used), all extrapartition and intrapartition transient data messages are included.

**Example**

To specify that no transient data messages from any queue will be eligible for processing by the Rule Processor:

```
EXTRA=( NO ) , INTRA=( NO )
```

**Example**

To specify that only intrapartition transient data messages will be eligible for processing by the Rule Processor:

```
EXTRA=( NO ) , INTRA=( YES , INCLUDE )
```

The CICS transient data name table will be an inclusion table for all intrapartition transient data messages from all queues.

**Example**

To specify that only intrapartition transient data messages will be eligible for processing by the Rule Processor:

```
EXTRA=( NO ) , INTRA=( YES , EXCLUDE )
```

The CICS transient data name table will be an exclusion table for intrapartition transient data messages. In this example, since the QUEUES= parameter is not used, no intrapartition queues are explicitly excluded. Thus, all intrapartition queues are included.

---

# Chapter 13 Implementing MAINVIEW Products in IMS

This chapter describes how to install and implement BMC Software modules, which are required for

- MAINVIEW AutoOPERATOR for IMS
- MAINVIEW for IMS (MVIMS) Online

Follow the procedures in this chapter if you are customizing one or both of these products and you have not performed AutoCustomization. AutoCustomization tailors your products automatically.

You can refer to the manual customization steps in this chapter if you need help during AutoCustomization.

**Note:** If you have more than one IMS, you might want to allocate a BBPARM data set that is unique to a IMS, as described in “Setting Up BBPARM Data Sets” on page 13-3 and “Modifying the IMS Control Region JCL” on page 13-6.

For information about how to use product libraries, including parameter libraries (BBPARM and UBBPARM) and sample libraries (BBSAMP and UBBSAMP), see “Using MAINVIEW Product Libraries” in the *MAINVIEW Common Customization Guide* or “Using Product Libraries” in the *MAINVIEW Administration Guide*.

## Deleting Modules from a Prior Release

You can skip this section if you are installing MAINVIEW AutoOPERATOR for IMS for the first time.

If a previous release of MAINVIEW for IMS, MAINVIEW for DBCTL, or MAINVIEW AutoOPERATOR for IMS was installed on the target IMS, use the information in Table 13-1 and follow the instructions below.

If you copied BBLINK members to an authorized STEPLIB data set using ICOPY, you can use the following BBSAMP jobs to delete the old modules:

**Table 13-1 BBSAMP Jobs to Delete Old Modules**

<b>IMS Release</b>	<b>MVIMS 3.1 MVDBC 2.1 AO 3.1, AO 4.1</b>	<b>MVIMS 3.2 MVDBC 3.2 AO 4.1</b>	<b>MVIMS 3.3.xx MVDBC 3.3.xx AO 5.1, AO 6.1, AO 6.2, AO 6.3</b>
IMS 5.1	IDEL31\$5	IDEL32\$5	IDEL33\$5
IMS 6.1	n/a	IDEL32\$6	IDEL33\$6
IMS 7.1	n/a	n/a	IDEL33\$7

To delete the old modules:

- Step 1** From Table 13-1, select the delete job that corresponds to the BMC Software product (or products) and IMS release installed at your site.
- Step 2** Edit the delete job you selected to change all &RESLIBs to the name of the data set where the old modules reside.
- Step 3** Run the delete job.

If you added BBLINK to the IMS STEPLIB concatenation, replace the BBLINK data set in the current IMS STEPLIB concatenation with the new BBLINK data set.

If you included BBLINK in the link list concatenation, replace the BBLINK data set in the current link list concatenation with the new BBLINK data set. This data set replacement will affect all the IMS address spaces running in that OS/390.

## Setting Up BBPARM Data Sets

The customization instructions in this chapter refer to the following data sets:

*hilevel.ibbparm*  
*hilevel.UBBPARM*  
*hilevel.BBPARM*

*hilevel*

The high-level data set name qualifier used at your site

*ibbparm*

A user-defined parameter data set unique to this IMS

You can allocate a separate *ibbparm* data set to contain any members that you want to make unique to the IMS, such as IMFSYS00 and IMFECPO0. The *ibbparm* data set must be allocated; it is not created through AutoCustomization.

**Note:** If the only members that require customization for an IMS are IMFSYS00, IMFECPO0, or both, an *ibbparm* data set is not required. Instead, you can create a renamed (*imsidSYS*) version of the IMFSYS00 member, a renamed (*imsidECP*) version of the IMFECPO0 member, or both, include them in UBBPARM, and customize them to suit your needs (as explained below in UBBPARM).

UBBPARM

A parameter data set that is tailored from the distributed BBPARM data set and is shared by all IMS systems

If you used AutoCustomization, you can use the UBBPARM data set created by AutoCustomization. If you did not use AutoCustomization, allocate UBBPARM, copy the distributed BBPARM data set to it, and tailor UBBPARM to suit your needs.

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 ID code)

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

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

If you do not need to customize *other* UBBPARM members for an IMS system, you do not need to create and allocate an *ibbparm* member for that system.

## BBPARM

The target BBPARM data set distributed by BMC Software

**Note:** Throughout the MVA AO books, parameter library members are normally referred to as BBPARM members, even though customized versions of the members may reside in the UBBPARM data set or in an IBBPARM data set.

The system parameter member is normally referred to as BBPARM member IMFSYS00, and the Event Collector parameter is normally referred to as BBPARM member IMFECPO0, even though the members may reside in a parameter data set with a member name in the format *imsidSYS* or *imsidECP*.

# Customizing BBPARM Members for an IMS System

This section is provided primarily for new installations of MAINVIEW AutoOPERATOR for IMS. If you have a prior version installed, you can use your existing parameter settings, but this would be a good time to review them.

This following procedure customizes the parameter members required to establish communication between an IMS region and the BBI-SS PAS

Set the system parameters in an *imsidSYS* member in the UBBPARM data set or in an IMFSYS00 member in an *ibbparm* data set. The parameters and their uses are as follows:

- SUBSYS.** Use this parameter to identify the subsystem name of the BBI PAS that the IMS region should communicate with. The subsystem should be the same as the one specified in the BBIJNT00 member of the UBBPARM data set or an *ibbparm data set*.
- MSGLVL1.** Set the message level to MTO, WTO, BOTH, or NONE.
- KEYWARN.** Use this parameter to set a minimum number of days before expiration warning messages are issued for product keys. The default is 45 days. (The KEYWARN parameter applies to MVIMS Online, MVIMS Offline, and MAINVIEW for DBCTL only, and it is used only in the IMS control region, not the BBI-SS PAS.)
- AOEINIT.** This parameter is used with the DFSAOE00 initialization call. (See BBPARM member IMFSYS00 for more information.)

**AOEEXIT and AOIEXIT.**

Use these parameters to specify the names of user-written AO exit routines. You can also use AOEEXIT and AOIEXIT to specify the order in which AO exit routines get control and whether a return code is padded. See “Enabling AO Exit Routines” on page 13-7 for more information.

As distributed, IMFSYS00 activates all MVIMS components in IMS. You can use *imsidSYS* in the UBBPARAM data set or IMFSYS00 in an *ibbparm* data set to temporarily deactivate one or more components. BBPARAM member IMFSYSBB contains information about how to deactivate components. Copy what you need from IMFSYSBB to your UBBPARAM *imsidSYS* member or *ibbparm* IMFSYS00 member.

## Activating MAINVIEW AutoOPERATOR for IMS

To activate MAINVIEW AutoOPERATOR for IMS, you need to enable product initialization and product authorization.

### Enabling Product Initialization

To enable MAINVIEW AutoOPERATOR for IMS initialization, in BBPARAM member BBISSP00, specify

**PRODUCT=AAO**

### Enabling Product Authorization

You must use product keys to enable MAINVIEW AutoOPERATOR for IMS product authorization.

#### Product Keys

If you have licensed MAINVIEW AutoOPERATOR for IMS and you want to use a product key to activate the product, specify product key IAO in BBPARAM member BBKEYS.

## Modifying the IMS Control Region JCL

The following sections describe how to allocate a BBPARM data set and provide IMS access to the BBLINK modules.

### Allocating a BBPARM Data Set

To establish the parameters for MVIMS Online and MAINVIEW AutoOPERATOR for IMS, allocate the BBPARM data sets by adding the following statements to the IMS control region startup procedure:

```
//IMFPARM DD DSN=hilevel.ibbparm  
//          DD DSN=hilevel.UBBPARM  
//          DD DSN=hilevel.BBPARM
```

### Giving IMS Access to MAINVIEW AutoOPERATOR for IMS

The IMS control region must be able to access the BBLINK modules to enable execution of MVIMS Online and MAINVIEW AutoOPERATOR for IMS.

If the BBLINK data set is in the LNKST concatenation, MAINVIEW AutoOPERATOR for IMS access is already established. If not, you can provide access by modifying your IMS control region JCL or by copying the required modules to the site authorized library (such as IMS RESLIB).

To modify the IMS control region JCL, add the BBLINK data set to the IMS STEPLIB concatenation. (BBLINK must be authorized.)

To copy the modules used in IMS, copy the individual BBLINK members to an authorized STEPLIB data set. Select one of the following jobs in BBSAMP:

- ICOPY8 for IMS 5.1
- ICOPY9 for IMS 6.1
- ICOPY1 for IMS 7.1

Then edit the JCL and run the job. This job copies the appropriate BMC Software modules from the BBLINK library to the site-authorized library, such as IMS RESLIB. You must rerun the JCL each time you apply BMC Software service.

## Enabling AO Exit Routines

You can skip this section if you have a prior version of MAINVIEW AutoOPERATOR for IMS installed.

IMS gives control to AO exit routines to do initialization processing and to do message processing. BMC Software does not supply an AO exit routine to do initialization processing. If you have your own routine, specify its one- to eight-character load module name in the AOEINIT parameter in an *imsidSYS* member in the UBBPARM data set or an IMFSYS00 member in an *ibbparm* data set.

BMC Software supplies two AO exit routines to do message processing: a type-1 AO exit routine, DFSAOUE0, and a type-2 AO exit routine, DFSAOE00.

**Note:** See the IBM *IMS Operations Guide* for an explanation of how the DFSAOUE0 and DFSAOE00 routines differ.

The DFSAOUE0 and DFSAOE00 routines capture MTO messages and IMS commands and pass them to MAINVIEW AutoOPERATOR for IMS. They also perform the following functions:

- provide an interface with your AO exit routines (if any)
- automatically start up MVIMS monitors when IMS starts up
- provide an interface between the Event Collector and the BBI-SS PAS for workload data collection
- initialize the Event Collector

On entry to DFSAOE00 with AOE0FUNC=1, MVIMS loads and executes any user exit specified in AOEINIT=xxxxxxx one time only. The DFSAOE00 exit does not forward control to the user's DFSAOE01 exit on the initialization call.

If you require a DFSAOE00 or DFSAOUE0 exit routine in addition to the exit provided by BMC Software, perform Step 1 and Step 2 below.

1. If you have
  - A. **One DFSAOE00 exit:** Rename it DFSAOE01, or choose a different name and use the control statements described in Step B.
  - B. **Multiple DFSAOE00 exits in addition to the BMC Software-supplied exit, or one exit not named DFSAOE01:** You must add one or more of the following AOEXIT control statements to an *imsidSYS* member in UBBPARM or an IMFSYS00 member in an *ibbparm* data set.

For example:

```
AOEXIT=MYNAME  
AOEXIT=DFSAOE02
```

In the example above, both exits are loaded and executed by the BMC Software DFSAOE00 exit. Program MYNAME executes first.

2. If you have
  - A. **One DFSAOUE0 exit:** Rename it DFSAOUE1, or choose a different name and use the control statements described in Step B.
  - B. **Multiple DFSAOUE0 exits in addition to the BMC Software-supplied exit, or one or more exits not named DFSAOUE1:** You must add one or more of the following AOIEXIT control statements to the *imsidSYS* member in UBBPARM or in an IMFSYS00 member in an *ibbparm* data set.

For example:

```
AOIEXIT=MYNAME  
AOIEXIT=DFSAOUE2
```

In the example above, both exits are loaded and executed by the BMC Software DFSAOE00 exit. Program MYNAME executes first.

## Flow of Control

By default, the BMC Software DFSAOE00 exit invokes the BMC Software AO exit routine (IELOAD) before it invokes your user exits. The return code set by the last user exit processed is passed to IMS. You can change the default processing in an *imsidSYS* member in the UBBPARM data set or an IMFSYS00 member in an *ibbparm* data set by specifying

- the order that exits are to be processed
- the exit return code

For example:

```
AOEEXIT=( DFSAOE01 , RC )  
AOEEXIT=IELOAD
```

In the example above, the DFSAOE01 user exit assumes control before the IELOAD exit and the DFSAOE01 return code is passed to IMS.

IMS then executes the DFSAOUE0 exit unless indicated otherwise by the return code set by the DFSAOE00 exit. The BMC Software DFSAOUE0 exit calls your exits in the order you specified with the AOIEXIT control statements. By default, the return code set by the last user exit processed is passed to IMS. You can pass the return code from another exit by specifying the RC parameter with the AOIEXIT control statement as shown in the following example:

```
AOIEXIT=( USEMINE , RC )  
AOIEXIT=IGNOREME
```

## Enhancement to the IMS DFSAOE00 Interface

BMC Software enhances the programming interface when your DFSAOE00 exit is invoked in the following ways:

- Register 11 contains the address of the IMS SCD.
- Register 13 contains the address of 15 prechained save areas.

## MAINVIEW AutoOPERATOR for IMS Considerations

When you install your user exit with a BMC Software exit, you should consider the following information:

- If the BMC Software exit is executed before the user exit and the user exit's return code cancels the processing of additional message segments of a multi-segment message, the message segments are also canceled for MAINVIEW AutoOPERATOR. The result is that incomplete IMS messages are passed to the MAINVIEW AutoOPERATOR and LAST SEG LOST messages from BBI. The MAINVIEW AutoOPERATOR AO exit holds the first segment for a certain length of time while waiting for additional segments, which can cause the messages to be processed out of time stamp sequence by MAINVIEW AutoOPERATOR.
- When the user exit is executed before the BMC Software exit, changes to the messages made by the user exit are received by the MAINVIEW AutoOPERATOR AO exit.
  - If the user exit sets the length code of a message (or a segment of a multi-segment message) to zero, MAINVIEW AutoOPERATOR does not process the message (or segment).
  - If the user exit sets the length code of the first segment to 0 and the return code to 4, all further segments are canceled for MAINVIEW AutoOPERATOR also.

**Note:** If any modules are specified with the AOIEXIT parameter in IMFSYS00 or *imsidSYS*, but the BMC Software exit is not specified, the exit still executes after all specified exits.

The MAINVIEW AutoOPERATOR AO exit and any other AO exit you use are under ESTAE protection when invoked by the BMC Software routines. If an abend occurs in one of these exits, only that routine is disabled; the other routine and IMS itself are not affected.

# Stopping Message Queuing to the IMS Master Terminal

Two methods of stopping message queuing to the IMS master terminal are available:

- Physically turn off the master terminal (this requires BBI Logging).

MAINVIEW AutoOPERATOR detects that the master terminal is not operable and will cancel any messages that are queued.

- Create a MAINVIEW AutoOPERATOR IMS-initiated Rule to suppress messages from the master terminal. Refer to the section on Rules in the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

If the master LTERM is defined as NON-SNA (in other words, as a local 3270), IMS **does not** indicate to the AOI exit that the terminal is inoperable. To avoid this problem, you can

- change the definition of the master LTERM
- prevent the master LTERM from connecting to VTAM
- dynamically force the VTAM node inactive by issuing the command **V NET,ID=nodename,INACT,FORCE**



---

---

# **Chapter 14 Implementing MAINVIEW AutoOPERATOR Access NV**

This chapter describes how to install the MAINVIEW AutoOPERATOR Access NV option into NetView and how to define the default NetView system to MAINVIEW AutoOPERATOR.

## Overview

To implement the MAINVIEW AutoOPERATOR Access NV option for NetView and define the default NetView system to MAINVIEW AutoOPERATOR, you must complete the steps in the following table. When the steps are described in this chapter, the page number of the discussion appears in the right column.

Task Description		Page Number
_____	modify NetView JCL	14-3
_____	create member NAIOPT00 in the NetView DSIPARM data set	14-4
_____	make backup copies of DSIPARM members DSICMD, DSIDMN, and DSIOPF	N/A <sup>a</sup>
_____	modify DSIPARM member DSICMD	14-4
_____	modify DSIPARM member DSIDMN	14-5
_____	copy members NAIINIT and NAISVAR into the NetView DSICLD data set	14-6
_____	modify member DSIOPF in DSIPARM	14-6
_____	verify operator profiles	14-6
_____	review NetView security for Access NV operator IDs	14-7
_____	review use of DSIEX02A exit	14-8
_____	review use of DSIEX14 exit	14-8
_____	modify member BBISSP00 in the BBPARM data set	14-9
_____	restart NetView to implement JCL and parameter changes	N/A <sup>a</sup>
_____	restart the BBI-SS PAS to implement parameter changes	N/A <sup>a</sup>

<sup>a</sup> These steps do not have individual descriptions in this chapter.

**Note:** If NetView is started before JES (SUB=MSTR), and you want to obtain dumps from the NetView product address space, you must complete the following steps:

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

**Step 2** Add the following allocations to your NetView address space JCL, pointing to the dump data sets created in the previous step.

```
//BBDUMP01 DD DISP=SHR,DSN=dumpdsn1,VOL=SER=...,UNIT=...
//BBDUMP02 DD DISP=SHR,DSN=dumpdsn2,VOL=SER=...,UNIT=...
//BBDUMP03 DD DISP=SHR,DSN=dumpdsn3,VOL=SER=...,UNIT=...
```

**Step 3** Catalog all data sets referenced in the NetView address space JCL PROC in the master catalog or change all allocations to UNIT/VOLSER allocations.

## Modifying NetView JCL

To make MAINVIEW AutoOPERATOR programs and parameters available in the NetView address space, modify the NetView JCL:

**Step 1** Add the MAINVIEW AutoOPERATOR BBLINK library to the NetView STEPLIB concatenation.

**Step 2** Add the following new DD statements to the NetView JCL:

```
//BBIPARM DD ... DSN=hilvl.BBPARM,DISP=SHR
//BBILOAD DD ... DSN=hilvl.BBLINK,DISP=SHR
```

## Creating DSIPARM Member NAILOPT00

Create member NAILOPT00 in the NetView DSIPARM data set to define:

- BBI-SS PAS with which NetView will communicate
- OSTs used to execute IMFEXEC NetView commands

**Step 1** Copy MAINVIEW AutoOPERATOR BBSAMP member NAILOPT00 to the NetView DSIPARM data set.

**Step 2** Edit NAILOPT00, changing the SSID parameter to specify the subsystem ID of the BBI-SS PAS that you want to process NAIEXEC commands issued from NetView EXECs.

**Note:** NAIEXEC commands can be sent to a different BBI-SS PAS using the TARGET keyword on the command. However, all NAIEXEC commands are initially routed to the specified BBI-SS PAS and then, if necessary, routed to another BBI-SS PAS using SS PAS-to-SS PAS communication.

**Step 3** Add 19 (or less) OPERxx statements to define the operator station tasks (OSTs) used to execute the IMFEXEC NETVIEW commands issued by MAINVIEW AutoOPERATOR EXECs.

You must specify at least one OPERxx statement; multiple statements do not need to be sequential.

The operator IDs specified should be used solely for Access NV. A user should not log on to NetView with any of these IDs.

## Modifying DSIPARM Member DSICMD

Modify the DSIPARM member DSICMD by appending the contents of BBSAMP member NAICMDS to the end of your existing DSICMD member.

This action adds definitions for the two NetView command processors, NAIEXEC and NAISTUB, used by Access NV.

Access NV uses two command processors that must be defined to NetView in the DSICMD member in the DSIPARM data set.

A preferred and more efficient alternative to using NAIEXEC to perform automation tasks from the NetView address space is available. This alternative is AOAnywhere, which offers more commands and does not use CSA storage. Additionally, AOAnywhere enables you to invoke from TSO or MAINVIEW AutoOPERATOR the same EXEC that are used in NetView.

To use AOAnywhere to perform automation from the NetView address space, you must add the command processor AOEXEC to the DSICMD member of the DSIPARM data set. If you attempt to execute AOEXEC without first adding AOEXEC to DSICMD, you will get an *invalid command* message from NetView. If you attempt to use ADDRESS LINKMVS to invoke AOEXEC a return code of 54 is issued. To add this command, specify the following code in DSICMD:

```
AOEXEC  CMDMDL  MOD=AOEXEC,TYPE=R,RES=N,ECHO=N
```

For more information about the functions of AOEXEC, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

## Modifying DSIPARM Member DSIDMN

Some Access NV functions run as an optional subtask (OPT) under NetView.

Modify the DSIPARM member DSIDMN to define this OPT by inserting the contents of BBSAMP member NAITASK before the END statement of your existing DSIDMN member.

This action adds a definition for the NetView optional subtask (OPT), NATASK, used by Access NV.

INIT=Y is included on the NATASK definition so that Access NV automatically initializes when NetView starts. If this parameter is modified, the task may also be started using the NetView START TASK=NATASK command.

The default NATASK definition does not have a priority specification. If high activity in the NetView address space prevents acceptable response time from Access NV, you might want to specify a priority for NATASK.

## Copying BBSAMP Members to the NetView DSICLD Data Set

To make the three CLISTs NAIINITX, NAIINIT, and NAISVAR, available to NetView, copy BBSAMP members NAIINITX, NAIINIT, and NAISVAR into the DSICLD data set.

## Modifying DSIPARM Member DSIOPF

The DSIOPF member in DSIPARM allows you to associate passwords and profiles with operator IDs.

Modify member DSIOPF in DSIPARM:

- Step 1** Add the new operator IDs that were previously specified in NAIOPT00.

The profile specified for these IDs must be similar to the example profile in BBSAMP member NAIPROFX. See the following section for more information.

- Step 2** If necessary, add operator IDs for TS users who will use Access NV.

The NetView operator ID for a TS user is the same as the BBI-TS ID. The profile specified for these IDs must be similar to the example profile in BBSAMP member NAIPROF. See the following section for more information.

## Verifying Operator Profiles

All Access NV controlled operator station tasks (OSTs) *must* execute an initialization CLIST that contains the NAISTUB command. This requirement applies to TS users as well as the OPERxx IDs used for IMFEXEC NETVIEW commands.

To ensure that NAISTUB is executed, follow these steps:

- Step 1** Copy BBSAMP members NAIPROF and NAIPROFX into the NetView DSIPRF data set.

- Step 2** If no profile specification already exists for an operator ID, specify either NAIPROFX or NAIPROF and use
- NAIPROFX for OPERxx IDs that were specified in NAILOPT00
  - NAIPROF for TS operator IDs
- Step 3** If a profile is already specified for an operator ID, browse the profile in the NetView DSIPRF data set to determine which initialization CLIST is being used.
- Step 4** Edit this CLIST in the NetView DSICLDA data set to include the statements from either NAINITX or NAINIT and use
- NAIINITX for OPERxx IDs that were specified in NAILOPT00
  - NAIINIT for TS operator IDs

## Reviewing NetView Security for Access NV Operator IDs

When NetView is accessed from a TS session, the NetView operator ID is the same as the TS user ID. Standard NetView security applies to the operator ID.

When NetView is accessed from a MAINVIEW AutoOPERATOR EXEC (IMFEXEC NETVIEW), the NetView operator ID is chosen by using a least-recently used algorithm from the pool of OPERxx IDs defined in DSIPARM member NAILOPT00. Standard NetView security applies to the operator ID.

All OPERxx IDs must have the maximum amount of authority required by any MAINVIEW AutoOPERATOR EXEC since a pool of operator IDs is used for processing IMFEXEC NETVIEW commands. In addition, these operator IDs must have authority to issue NetView START and STOP commands.

## Reviewing the Use of NetView Exit DSIEX02A

MAINVIEW AutoOPERATOR Access NV uses the NetView exit DSIEX02A. This exit processes line mode terminal and console output and allows messages to be modified.

If you do not currently have a DSIEX02A exit, no action is required to install the MAINVIEW AutoOPERATOR exit. If you already use this exit, do the following steps to make sure that your existing DSIEX02A exit and the MAINVIEW AutoOPERATOR Access NV DSIEX02A exit can coexist:

**Step 1** Rename your existing DSIEX02A load module to DSIEX02B.

**Step 2** Link the two exits by using the following linkage editor statements:

```
//OLDLIB DD DSN=the library where your renamed exit
resides
//SYSLMOD DD DSN=AOBBLINK
CHANGE DSIEX02A(DSIEX02B)
INCLUDE OLDLIB(DSIEX02B)
INCLUDE SYSLMOD(DSIEX02A)
ORDER DSIEX02A
ENTRY DSIEX02A
NAME DSIEX02A(R)
```

MAINVIEW AutoOPERATOR Access NV first passes control to DSIEX02B. If DSIEX02B has modified the incoming message, MAINVIEW AutoOPERATOR Access NV processes the modified message. If the exit indicates that the message should be discarded, MAINVIEW AutoOPERATOR Access NV returns control to NetView with a return code 4.

## Reviewing the Use of NetView Exit DSIEX14

MAINVIEW AutoOPERATOR Access NV uses the NetView exit DSIEX14. This exit intercepts LOGOFF requests and performs clean-up processing for Access NV controlled OSTs. LOGOFF requests for OSTs that are not controlled by Access NV are ignored.

If you already use this exit, follow the steps below to make sure that your existing DSIEX14 exit and the MAINVIEW AutoOPERATOR Access NV DSIEX14 exit can coexist:

**Step 1** Rename the existing DSIEX14 load module to DSIEX14B.

**Step 2** Link the two exits by using the following linkage editor statements:

```
//OLDLIB DD DSN=the library where your renamed exit
resides
//SYSLMOD DD DSN=AOBBLINK
CHANGE DSIEX14(DSIEX14B)
INCLUDE OLDLIB(DSIEX14B)
INCLUDE SYSLMOD(DSIEX14)
ORDER DSIEX14
ENTRY DSIEX14
NAME DSIEX14(R)
```

MAINVIEW AutoOPERATOR Access NV first passes control to DSIEX14B with Register 1 pointing to the User Service Block (DSIUSE). Upon return from DSIEX14B, DSIEX14 will process the LOGOFF request and then return to NetView.

## Modifying BBPARM Member BBISSP00

Add a NETVIEW= specification to BBPARM member BBISSP00. This defines the default NetView target for the BBI-SS PAS. The BBI-SS PAS must be restarted for the parameter change to take effect.

The default NetView target is used when initially displaying the Access NV work station and as the target for any IMFEXEC NETVIEW commands that do not specify a target. However, you can access other NetView systems by overtyping the NetView target on the Access NV workstation or specifying a job name on the IMFEXEC NETVIEW command.



---

# Chapter 15 Implementing AutoRULE

This chapter describes how to implement the optional MAINVIEW AutoOPERATOR AutoCustomization step, “Run AutoRULE for AutoOPERATOR, Event Management Utility”.

## Overview

AutoRULE is an optional advanced customization step for MAINVIEW AutoOPERATOR that allows you to automatically generate MAINVIEW AutoOPERATOR Rules from your existing Message Processing Facility (MPF) lists and AutoMate Rules.

You can also use the BMC Software-supplied KnowledgeBases to generate MAINVIEW AutoOPERATOR Rules. The generated Rules perform automation such as suppressing messages from the console, replying to write-to-operator messages (WTORs), and issuing OS/390 commands.

## Prerequisites

To use AutoRULE, you must first fill in the worksheet (refer to Appendix H, “AutoRULE Worksheet”). The worksheet should be used in conjunction with the AutoCustomization steps because you need the data from the filled-in worksheet to proceed through the AutoCustomization steps.

## Completing the AutoRULE AutoCustomization Step

The following describes the panels you must complete while executing the AutoCustomization step, “Run AutoRULE for AutoOPERATOR, Event Management Utility”. You need the information from the filled-in worksheet to respond to the AutoCustomization steps.

- Step 1** Select the step titled “Run AutoRULE for AutoOPERATOR, Event Management Utility” to implement AutoRULE from AutoCustomization.

Figure 15-1 is displayed.

**Figure 15-1 AutoRULE Event Management: Panel 1**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>

AutoRULE is an optional AutoOPERATOR event management utility. It allows
you to create and maintain AutoOPERATOR Rulesets. The Rulesets are built
using the supplied KnowledgeBases, input from MVS MPF lists, and AutoMate
Rules. AutoRULE can be run multiple times, but should be run at least once
each time you receive a KnowledgeBase.

All AutoRULE panels have context sensitive help and each step is optional.

Please ensure you have reviewed the AutoRULE documentation and fill in the
supplied worksheets prior to executing this step.

Would you like to run AutoRULE? at this time ==> YES (Yes/No) [1]

    If you reply NO, this step will be bypassed.

    If you reply YES, you will invoke AutoRULE Event Management

Press ENTER to continue, HELP for more information, or END to exit.
    
```

- Step 2** To proceed, enter **YES** at [1].

You are prompted to select the conversion process that you want to complete, as shown in Figure 15-2 on page 15-3.

**Figure 15-2 AutoRULE Event Management: Panel 2**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
OPTION ==>

Please complete the steps applicable to your site           User ID : BAOMXY1
in the order shown below. Each step may be rerun         Time   : 15:20
as often as needed before moving on to the next.         Date   : 96/04/25

                                                                Last Run on
                                                                -----
[2] M  MPF      Convert MPF List                          3 Apr 1996 13:07:34
[3] A  AUTOMATE Convert AutoMate Rules                    25 Apr 1996 13:03:09
[4] C  COMPARE  Compare & Delete Duplicates              10 Apr 1996 11:04:29
[5] R  RULES    Generate AutoOPERATOR Rules              25 Apr 1996 12:49:55
      O  OUTPUT  Browse Output

Press END to return, HELP for more information or select an OPTION

```

**Step 3** Convert your automation.

- To convert MPF automation to MAINVIEW AutoOPERATOR Rules, choose the step Convert MPF List [2].

or

- To convert AutoMate automation to MAINVIEW AutoOPERATOR Rules, choose the step Convert AutoMate Rules [3].

You need the information from section 1 of the worksheet to complete these steps.

You must choose and complete one of these steps before you complete any of the other steps presented on this panel.

**Step 4** After the conversion you choose is completed, select the step Compare and Delete Duplicates (see [4] in Figure 15-2).

Figure 15-3 on page 15-4 is displayed.

**Figure 15-3 AutoRULE Event Management: Compare Option**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>

Line commands: / to select a member to process, E to edit, or B to browse.

Note that highlighted members must be viewed before being selected.

MVS Console Suppression:      (Ind.)   MVS Console Actions:          (Ind.)
    MVS Conservative List      MVS WTOR Replies
    MVS Aggressive List        MVS Commands
    CICS Messages              CICS WTOR Replies
MVS Console Alerts:          Non-MVS Alerts:
    CICS Alerts                CICS TDQ Alerts
    MVS Alerts                 BBI Journal Alerts
    Production Job Alerts      IMS AOI Exit Alerts
    Tape Mounts & Errors       DB2 Alerts
MVS Command Handling:        Converted User Automation (created)
    MVS Command Rejection      MPF List           3 Apr 1996
    User Commands              AutoMate Rules     25 Apr 1996

    Other member name (optional):

Press END to return, HELP for more information or enter GO to proceed.
    
```

Use this panel to select AutoRULE KnowledgeBase members to be included in your newly generated Rule Set. Each of the members is for a specific type of message and action type.

Before selecting members, you can browse or edit a member. Use the B or E line command to do this. The **(Ind.)** column indicates which members have been browsed or edited.

When you edit a member, you can change the flags to create a different type of Rule. For example, you might want to delete some of the automatic replies from the OS/390 Commands & Replies member. For a list of flags and their meanings, refer to “Describing AutoRULE Flags” on page H-7.

**Step 5** After the Compare step is finished and you have reviewed the error messages that are displayed, press **Enter** to go to the next step.

**Figure 15-4 AutoRULE Event Management: Compare Option Completed**

```
AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>>

          Compare & sort has successfully completed.

                What to do now:

In any cases where there were conflicting Rules for an event, your
converted MPF or AutoMate Rule was placed ahead of the distributed
AutoRULE Rule.  If you have changed your mind after reviewing the
results of the Compare & Sort step (in member ARLISTC ), you can
delete your Rule and rerun this step.

If you are happy with the results, go on to the next step:

Next, run Rules.  This will generate AutoOPERATOR Rules from your
converted Rules and the distributed KnowledgeBase Rules.

Compare & Sort has also created a sorted list of messages that will
be handled by the new Rules (in the member ARSORTED ).  You can
print this member for future reference.

Press END or ENTER to continue
```

**Step 6** Follow the instructions listed on this panel.

After the compare and sort step is complete, proceed to the step Generate MAINVIEW AutoOPERATOR Rules (see [5] in Figure 15-2 on page 15-3).

The Generate MAINVIEW AutoOPERATOR Rules step generates MAINVIEW AutoOPERATOR Rules in Rule Sets that are named RULRUxxx, where xxx is a user-specified three-character suffix.

**Step 7** Specify the suffixes for the Rule Sets. Enter the information from Section 3 of the worksheet on this panel.

**Figure 15-5      AutoRULE Event Management: Naming Rule Sets**

```
AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>

Ruleset Names:                               (3-character suffix ID)

MSGs starting with A - G                     MSGs starting with H - L
MSGs starting with M - S                     MSGs starting with T - Z
Journal Events                               CICS TDQ Events
IMS Events                                   Command Events

Rules Author                                 (For Documentation Fields)
Display on Console                           YES or NO
Catch-all Rules                             ENABLED DISABLED SAVE or NONE

STCs to Exclude

Userids to Allow

Press Enter to continue, HELP for more information, or END to return.
```

**Step 8**    Press **Enter**.

Figure 15-6 is displayed.

**Step 9**    On this panel, enter the information from Section 4 of the worksheet.

**Figure 15-6 AutoRULE Event Management: Creating a Production Job**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>>

Define a production job, using the following Rule fields.
Job name . .                Userid . .                RACF Group
Job class . .              Accounting Info . . .

      Variable-name          Op      Variable-value          AND/OR

Press ENTER to continue, END or CANCEL to exit without creating Rules.
HELP for more information.

```

This panel specifies the parameters that tell AutoRULE how to recognize a production job, and this information is used for creating Rules for production job abends.

The bottom portion of the panel allows you to specify variable dependencies that function exactly as the Variable Dependencies panel does during MAINVIEW AutoOPERATOR Rules creation (refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide* for information about creating Rules).

**Step 10** Press **Enter**.

Figure 15-7 on page 15-8 is displayed.

**Step 11** On this panel, enter the information from Section 5 of the worksheet.

**Figure 15-7 AutoRULE Event Management: Specifying ALERT Queues**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>

Specify User IDs and Alert Queues for problem notification.

TSO User IDs for Notification      Operator Alert Queue Names
MVS Problems          KMZ1          MVS Alerts           MVS
CICS Problems        BAKMZ1         CICS Alerts          CICS
IMS Problems         BAKMZ1         IMS Alerts           IMS
DB2 Problems         BAKMZ1         DB2 Alerts           DB2
Network Problems    BAKMZ1         Network Alerts       NET
DASD Problems       BAKMZ1         Tape Alerts          TAPE
Unauthorized Ccmds  BAKMZ1         Batch Alerts         BATCH
Automation Problems CMF6          All Other Alerts     MAIN

Optional ParmS
Alerts Target Name          Leave blank for no Alert routing
Default Beeper Name                Elan only
Default Beeper Info
                                (continued)

Press ENTER to continue, END or CANCEL to exit without creating Rules.
HELP for more information.
    
```

This panel contains the parameters that tell AutoRULE how to direct ALERTs after they are created. The user IDs allow you to specify a TSO user ID to be notified when an ALERT is created. To specify more than one TSO user ID, separate the IDs with spaces.

**Step 12** Press **Enter**.

Rules are generated and Figure 15-8 on page 15-9 is displayed.

**Figure 15-8 AutoRULE Event Management: Successful Completion Panel**

```

AUTOOPERATOR ----- AutoRULE Event Management ----- CUSTOMIZATION
COMMAND ==>

                AutoRULE Rules Generation Successfully Completed

                What to do now:

If you didn't get the results you expected, you can go back and fix
the errors and rerun this or any other step.  The listing you just
viewed is in member ARLIST in the UBBPARM dataset.

UBBPARM member MPFNEW is a replacement version of MPF list.  After
you implement the new Rules, copy it to SYS1.PARMLIB with the same
name as the current MPF list (MPFLSTxx).  Then issue the MVS MPF
list reset command.

Use the OUTPUT option to aid in completing your Event Management
conversion process.

Press END or ENTER to continue

```

The new UBBPARM members that have been created by AutoRULE are

- ARLIST, which contains the results of the AutoMate conversion

If any AutoMate Rules were not converted by AutoRULE, you will have to convert them by manually creating MAINVIEW AutoOPERATOR Rules for them. Refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide* for information about creating MAINVIEW AutoOPERATOR Rules.

- MPFNEW, which contains a new listing that you can use to replace the existing MPFLSTxx member in SYS1.PARMLIB

To use the new member, make a copy of the existing MPFLSTxx member. Next, copy MPFNEW to SYS1.PARMLIB and rename it to the previous MPFLSTxx member name. Last, issue the MVS MPF list reset command.

- ARSORTED, which contains a list of the events from which the Rules were generated

You might want to print the contents of this member to have as a reference document of the events that now have Rules.

- Various new Rule Sets named RULRU $xxx$ , where  $xxx$  is a user-specified three-character suffix

To use these new Rule Sets, enter the new Rule Set names in BBPARM member AAOPRM $xx$  on the RULESET parameter.

The AutoCustomization step for AutoRULE is complete.

---

---

# Chapter 16 Implementing MAINVIEW AutoOPERATOR ALERT Interface to PATROL EM

This chapter describes how to customize MAINVIEW AutoOPERATOR to send ALERTs to the PATROL Enterprise Manager (PATROL EM) outboard processor.

For information about receiving MAINVIEW AutoOPERATOR ALERTs on the PATROL EM workstation, refer to the PATROL EM user documentation.

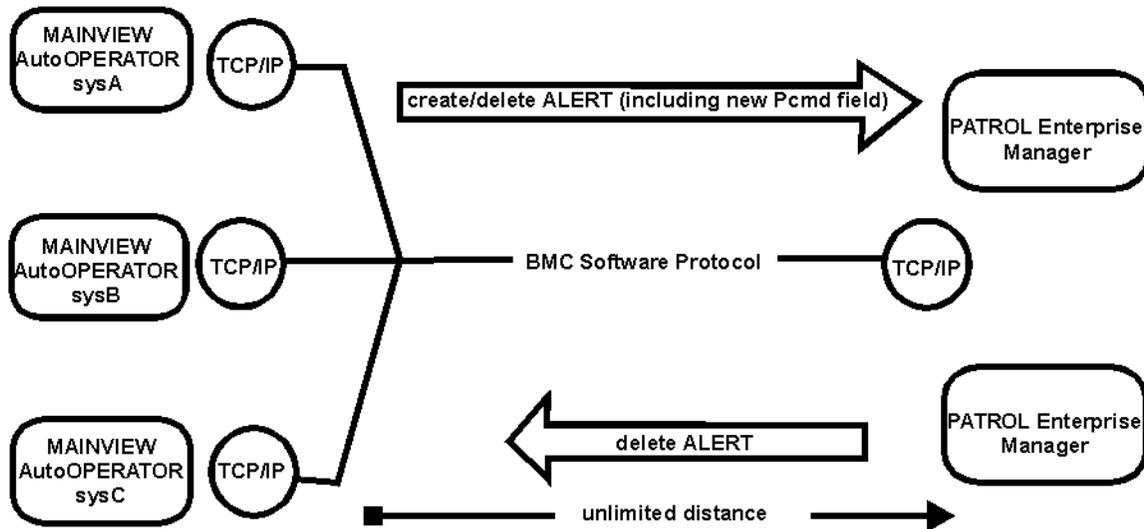
## Overview

You can implement the sending of ALERTs from MAINVIEW AutoOPERATOR to PATROL EM in the following ways:

- by implementing a TCP/IP-based interface called AO Connection (previously ConnectAAO) between MAINVIEW AutoOPERATOR and PATROL EM (see Figure 16-1 on page 16-2)
- by establishing a VTAM connection called ConnectOS390 to PATROL EM by using a unique ACB and an I/O Concepts X-Direct protocol converter (see Figure 16-2 on page 16-3)

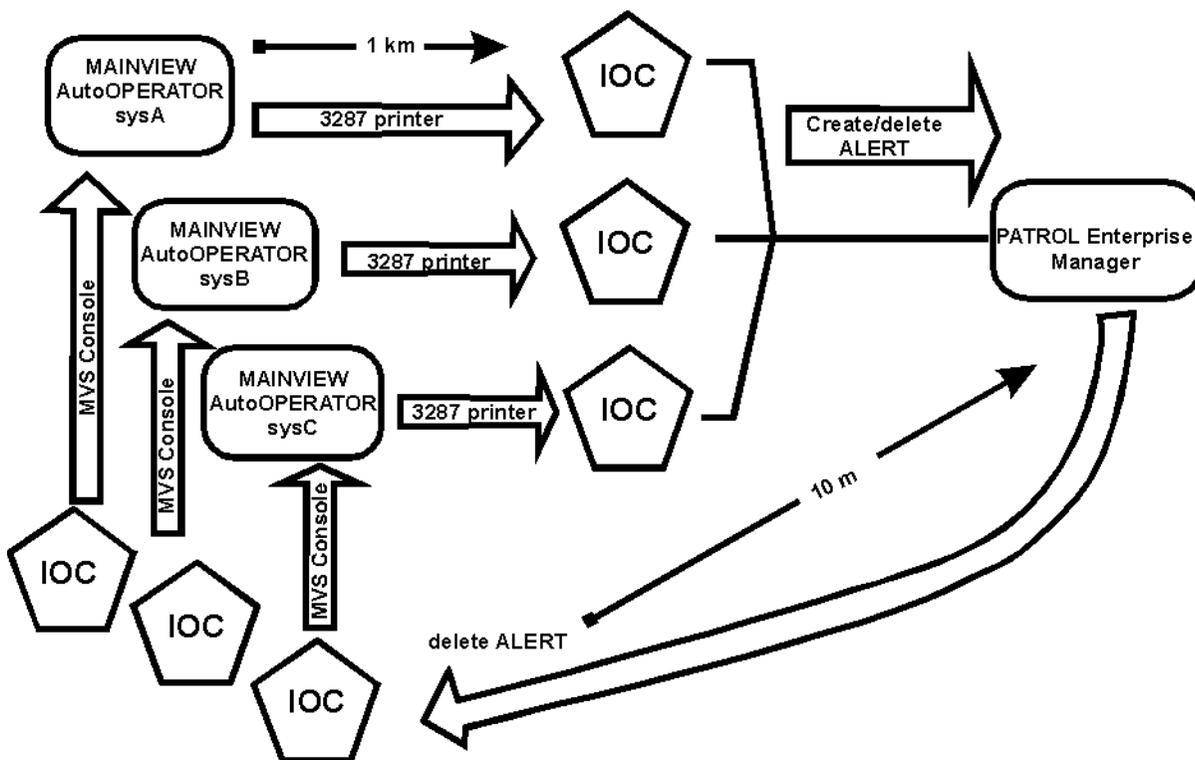
The following figures show the different configurations when MAINVIEW AutoOPERATOR is enabled to send ALERTs to PATROL EM.

Figure 16-1 MAINVIEW AutoOPERATOR and PATROL EM AO Connection Using TCP/IP



Note that in this configuration, no distance limit exists between the locations of the PATROL EM systems and the MAINVIEW AutoOPERATOR systems. Figure 16-2 on page 16-3, shows significant physical distance limitations.

Figure 16-2 MAINVIEW AutoOPERATOR and PATROL EM ConnectOS390 through I/O Concepts X-Direct Protocol Converters



Note that in this configuration, a limit of 1 km exists between MAINVIEW AutoOPERATOR and the I/O Concepts protocol converter, and a 10 m limit exists between PATROL EM and MAINVIEW AutoOPERATOR. Therefore, MAINVIEW AutoOPERATOR cannot send ALERTs if it is located more than 1 km away from the /O Concepts protocol converter, and PATROL EM cannot send ALERTs if it is located more than 10 m away from MAINVIEW AutoOPERATOR.

For more information about these two connection limitations, refer to

- “Establishing a TCP/IP Connection” on page 16-4
- Appendix K, “Establishing a VTAM Connection for MAINVIEW AutoOPERATOR to PATROL EM Communication”

**Note:** This feature requires that you install the latest version of the AO Connection Compatibility Patch.

## Establishing a TCP/IP Connection

The following sections describe how to enable MAINVIEW AutoOPERATOR to share ALERTs with PATROL EM through an AO Connection TCP/IP connection.

### What You Need

To establish a TCP/IP connection between MAINVIEW AutoOPERATOR and PATROL EM, you must have

- at least one BBI-SS PAS where MAINVIEW AutoOPERATOR 6.1 or later is running
- at least one PATROL EM system that has been configured to accept ALERTs from MAINVIEW AutoOPERATOR 6.1 or later

For information, refer to the PATROL EM AO Connection documentation.

- MVS/ESA TCP/IP 3.2 or later or Interlink TCPAccess 5.2 or later installed in the MVS system

You must also perform the following steps:

**Step 1** In BBPARM member BBISSP00, specify

**GTS=YES**

If default TCP specifications are insufficient for your installation, specify a two-character qualifier instead of YES and create a BBPARM member BBTTCP00. (Refer to “BBPARM Member BBTTCP00” in the *MAINVIEW Common Customization Guide* for more information about the BBTTCP00 member.)

**Step 2** This step can be done through the AutoCustomization step, “Customize MAINVIEW AutoOPERATOR ALERTs Publishing” or through the manual customization step that is documented in “Defining TCP/IP Addresses in MAINVIEW AutoOPERATOR” on page 16-5.

Create a BBPARM member AAOGME00 to identify the PATROL EMs that are to be connected. This member name can be overridden by using the AAOGME=xx card in the BBCFG file, where xx is the suffix to use for the AAOGME member. See Appendix E, “BBPARM Data Set Members for MAINVIEW AutoOPERATOR” for complete details about this member.

The AAOGME00 member allows MAINVIEW AutoOPERATOR to use the General Messages Exchange (GME) facility to share MAINVIEW AutoOPERATOR ALERTs with PATROL EM. The local GME node requires network addresses and port numbers to connect to remote GME nodes.

**Step 3** Specify

**GMEACT=YES**

**Step 4** Specify

**LSTNPORT=#####**

where ##### is a unique TCP port on this z/OS LPAR that is not in use by any other application. The same port can be used only when MAINVIEW AutoOPERATORS are on different z/OS LPARs. Use the TSO NETSTAT A command to determine which ports are currently in use.

**Step 5** Create unique TGTIP= entries for each PATROL EM AO Connection that is to be connected to this MAINVIEW AutoOPERATOR. See Appendix E, “BBPARM Data Set Members for MAINVIEW AutoOPERATOR” for a description of the BBPARM member AAOGME00 parameters beginning with TGT. They are the parameters that define a connection.

**Step 6** Reset the BBPARM member AAOGME00 using the BBI command

**.RESET PARM AAOGMExx RESYNC**

## Defining TCP/IP Addresses in MAINVIEW AutoOPERATOR

**Note:** If you completed the AutoCustomization step “Customize MAINVIEW AutoOPERATOR Alerts Publishing”, you do not have to complete the steps in this section.

To define TCP/IP addresses, follow these steps:

**Step 1** Create a member in your UBBPARM data set named AAOGME $xx$  where  $xx$  is a two-character suffix associated with the BBI-SS PAS that you are enabling to share ALERTs with PATROL EM.

**Step 2** (mandatory) Specify

**TGTIP=IP\_ADDRESS**

where **IP\_ADDRESS** is the name of the location of a server to which GME will connect.

GME allows IP addresses to be expressed in

- IP v4 (dotted decimal)
- IP v6
- Internet host domain name; for example:  
**www.remote\_host\_domain\_name.com**

If a host domain name is used, it is your responsibility to define that name in a domain name server (DNS).

**Step 3** (optional) Specify

**TGTPORT=xxxxx**

where *xxxxx* is a one- to five-digit decimal number specifying the port number on which the PATROL EM server is listening. The GME facility attempts to connect to the remote GME node using this port number. The default port number is 7051.

### Specifying More Than One TCP/IP Address

Note that you can specify as many TGTIP and TGTPORT parameters in BBPARM member AAOGME00 as you need. Any parameters set prior to the first TGTIP parameter are attributed to the local GME node.

The TGTIP parameter denotes definitions for remote GME nodes and the TGTPORT parameter should appear in column 1. Each subsequent TGTIP parameter signifies settings for a different remote GME node. For example:

```
TGTIP=137.72.4.15 ,TGTPORT=12345
```

```
TGTIP=172.18.14.255 ,TGTPORT=67890
```

Additional parameters can be set for each TGTIP address, each separated by a comma. Settings can continue onto additional lines; for example:

```
TGTIP=137.72.4.15,TGTPORT=12345,
TGTNAME=remote_GME_node_name_here,

TGTHB=999
```

For information about additional parameters, refer to “BBPARM Member AAOGME00 Parameters (MAINVIEW AutoOPERATOR Only)” on page E-16.

## Controlling the MAINVIEW AutoOPERATOR PATROL Enterprise Manager Interface

Table 16-1 shows BBI control commands that you can use to control the MAINVIEW AutoOPERATOR and PATROL EM interface.

**Table 16-1 BBI Control Commands for MAINVIEW AutoOPERATOR and PATROL EM Interface for TCP/IP Connections (Part 1 of 2)**

Command	Keyword	Parameter	Description
DISPLAY   D	GME   G	CONN   C PUBLISH   P RECEIVE   R TRACE   T TRACE,nnnn   T,nnnn no parm	Provides a summary status of the local GME node. <ul style="list-style-type: none"> <li>• CONN displays status of GME connections. A specific connection is displayed by specifying the GME NAME.</li> <li>• PUBLISH displays details of PUBLISHED GME subjects.</li> <li>• RECEIVE displays details of the GME applications that are registered to receive messages.</li> <li>• TRACE displays a summary of the trace records by class.</li> <li>• TRACE,nnnn displays up to 10 trace records starting with the specified trace record sequence number.</li> <li>• no parm displays status of the local GME node.</li> </ul>

**Table 16-1 BBI Control Commands for MAINVIEW AutoOPERATOR and PATROL EM Interface for TCP/IP Connections (Part 2 of 2)**

Command	Keyword	Parameter	Description
RESET   E	GME   G	nodename	<p>Stops and starts communications with a GME server node using the parameters specified from the active BBPARM member AAOGMExx.</p> <p>Possible specifications for nodename are</p> <p><b>TGTNAME</b> Is the name of the GME server node whose connection is to be stopped and started. Typically, this value will be the workstation ID of the system where the remote GME server node is running.</p> <p><b>Tgtip:Tgtport</b> Is a combination of the IP address and the port number connected by a colon (:).</p>
RESET   E	PARM   P	AAOGMExx   RECYCLE AAOGMExx   RESYNC	<p>Reinitializes the parameters specified by AAOGME00.</p> <p>Possible options are to specify:</p> <p><b>RECYCLE</b> Stops the GME environment, disabling all connections, and restarts the GME environment, re-establishing all connections.</p> <p><b>RESYNC</b> Stops any active GME connections that might exist (but have been removed from the active BBPARM member AAOGMExx) and starts any new connections that may have been added to AAOGMExx since the recycle or startup of MAINVIEW AutoOPERATOR.</p>
START   S	GME   G	nodename	<p>Starts communications with a GME server node using connection parameters read from the active BBPARM member AAOGME00.</p> <p>For a description of possible values for nodename, refer to the description for RESET GME nodename in this table.</p>
STOP   P	GME   G	nodename	<p>Stops communication with a GME server node.</p> <p>For a description of possible values for nodename, refer to the description for RESET GME nodename in this table.</p>

For a description of all the available BBI control commands, refer to the *MAINVIEW Administration Guide* where the complete list of BBI control commands is documented.

# How MAINVIEW AutoOPERATOR and PATROL EM Share ALERTs

The following sections describe how

- MAINVIEW AutoOPERATOR and PATROL EM synchronize their ALERT displays
- ALERTs and ALERT queues are created and deleted between the two products
- PATROL EM issues commands through MAINVIEW AutoOPERATOR

after the connection between MAINVIEW AutoOPERATOR and PATROL EM is completed.

## ALERTs Synchronization

MAINVIEW AutoOPERATOR synchronizes its set of ALERTs with the PATROL EM set of ALERTs when a connection has been established.

The process of synchronizing is as follows:

1. MAINVIEW AutoOPERATOR initializes.
2. MAINVIEW AutoOPERATOR sends ALERTs destined for PATROL EM.
3. PATROL EM deletes any ALERTs that were deleted in MAINVIEW AutoOPERATOR while it was not connected to PATROL EM.
4. PATROL EM deletes any ALERTs that were closed using the PATROL EM automation `exec closeAOCalert` while it was not connected to MAINVIEW AutoOPERATOR.

## Creating and Deleting ALERT and ALERT Queues

The following table describes what happens on the PATROL EM side when MAINVIEW AutoOPERATOR creates or deletes an ALERT destined for PATROL EM.

When	Then
an MAINVIEW AutoOPERATOR Rule (or EXEC) generates a PUBLISH(ADD) ALERT destined for PATROL EM	PATROL EM creates the ALERT. Existing ALERTs with an identical queue/key remain on PATROL EM but are not associated.
an MAINVIEW AutoOPERATOR Rule (or EXEC) generates an ALERT to <i>replace</i> an existing ALERT on PATROL EM	PATROL EM deletes all existing ALERTs that match the queue/key for that AutoOPERATOR. PATROL EM creates the ALERT.
an MAINVIEW AutoOPERATOR Rule (or EXEC) deletes an ALERT or ALERT queue that resides on PATROL EM	PATROL EM deletes the specified ALERT or deletes all ALERTs it received from the MAINVIEW AutoOPERATOR GME node that belong to the specified queue.
PATROL EM executes automation script closeAOCalert for an ALERT	MAINVIEW AutoOPERATOR deletes the specified ALERT.

---

---

# Chapter 17 Implementing MAINVIEW AutoOPERATOR Advanced JES3 Support

Unlike JES2, JES3 manages its own message traffic and command execution internally and may use dedicated consoles to interact with the operator.

Depending on the JES3 release, these messages and commands might never become visible to the subsystem interface which MAINVIEW AutoOPERATOR uses to drive the Rule Processor. Note that no additional customization is required to process OS/390 messages and commands. Therefore, some additional customization steps are necessary in order to fully manage a JES3 environment and perform automation tasks with Rules on JES3 messages and commands.

JES3 releases prior to 2.2 must be handled differently from later versions. The following sections explain the necessary steps to implement full JES3 support.

## Using JES3 2.2 and Later

JES3 releases 2.2 and later use the standard OS/390 interface. However in most cases, JES3 will be the first subsystem to process messages and commands. For MAINVIEW AutoOPERATOR to correctly process JES3 messages and commands, MAINVIEW AutoOPERATOR must process the messages and commands before JES3 does.

To accomplish this, you must use the BBPARM member BBISSP00 parameter JESFLTR=. When this parameter is used, MAINVIEW AutoOPERATOR installs a JES filter that allows MAINVIEW AutoOPERATOR to see message or command traffic before JES does.

Possible values for the JESFLTR= parameter are as follows:

JESFLTR=WTO | CMD

or

JESFLTR=( CMD , WTO )

The following sections describe under what circumstances you can use these specifications.

## Intercepting Messages

Use JESFLTR=WTO to suppress messages from a JES3 console. This specification replaces the use of the IATUX31 exit, as described in “Intercepting Messages” on page 17-5.

JES3 messages are submitted to the Rules Processor with an event type of MSG before they are seen by JES.

## Intercepting Commands

Use JESFLTR=CMD to suppress messages from both JES2 and JES3 consoles. This specification affects

- all JES2 commands
- only JES3 commands that are entered at an OS/390 console

The JES3 command exit IATUX18 is required to suppress JES3 commands entered from a JES3 console.

JES3 commands are submitted to the Rules Processor with an event type of CMD.

## Implementing the JES Filter

You can implement the JES filter in either of the following ways:

- Through the JESFLTR parameter in BBPARM member BBISSP00; for example, specify

**JESFLTR=WTO**

to suppress messages from a JES3 console.

To suppress all JES2 commands and JES3 commands entered from an OS/390 console, specify

**JESFLTR=CMD**

To suppress both commands and messages, specify

**JESFLTR=(CMD,WTO)**

The filter is installed after you have made these specifications and the BBI-SS PAS is cold started.

- Through the BBI command .SET

Use this command to establish the filter if the BBI-SS PAS was started before JES or to reestablish the filter (if it was removed during the life of a BBI-SS PAS) by specifying

**.SET JESFLTR=WTO**

or

**.SET JESFLTR=CMD**

or

**.SET JESFLTR=(CMD,WTO)**

Issuing these commands will install the filter.

## Removing the Filter

To remove the filter at any time, regardless of whether the BBI-SS PAS is active or not, use the command UNHOOK. Enter the UNHOOK command as an OS/390 command and prefix it with the BBI-SS PAS command character prefix as defined by the CMDCHAR parameter in BBPARM member BBISSP00. The default command character is a period (.).

### Other Implementation Considerations

Do not restart JES while the filter is installed. If you restart JES while the filter is installed, one of two situations might happen:

- The JES restart might fail.
- MAINVIEW AutoOPERATOR might not be able to issue any JES messages or commands.

If either of these situations happen, follow these steps:

- Step 1** Issue the UNHOOK command from an OS/390 console.
- Step 2** Restart JES.
- Step 3** Reestablish the filter by either method described in “Implementing the JES Filter” on page 17-3.

## Issuing JES3 Commands from MAINVIEW AutoOPERATOR

To correctly issue JES3 commands, the JES3CMD= parameter in BBPARM member BBISSP00 should be set to JES3CMD=NO.

Full command response support is available.

## Using JES3 Version Prior to 2.2

Version of JES3 prior to 2.2 do not broadcast any information to the subsystem interface that MAINVIEW AutoOPERATOR uses to drive the Rule Processor.

MAINVIEW AutoOPERATOR provides a JES3 message exit (IATUX31) and a JES3 command exit (IATUX18) that are used to interact with the Rule Processor. Refer to BBSAMP members IATUX31 and IATUX18 for instructions about how to implement these required JES3 exits. By installing these exits, JES3 messages and commands can be seen by MAINVIEW AutoOPERATOR and automation (triggered by these messages and commands) with Rules is possible.

## Intercepting Messages

When the IATUX31 exit is installed, JES3 messages are used as input to the Rule Processor with an event type of JES3. OS/390 messages are sent twice to the Rule Processor: once from the Master Broadcast function (subsystem interface) with an event type of MSG and again from the IATUX31 exit with an event type of JES3.

Any Rule Processor actions that are against event type JES3 messages affect JES3 devices only. Any Rule Processor actions against event type MSG affect MCS consoles or the SYSLOG.

## Intercepting Commands

When the IATUX18 exit is installed, JES3 commands are received by the Rules Processor with an event type of CMD.

## Issuing JES3 Commands from MAINVIEW AutoOPERATOR

Prior to JES3 version 2.2, SVC 34 could not be used to issue JES3 commands, so the commands must be issued using the subsystem interface. To issue the commands, follow these steps:

- Step 1** Specify **JES3CMD=YES** in BBPARM member BBISSP00 to indicate that the subsystem interface should be used instead of SVC 34.
- Step 2** Specify the correct JES3 command character for your system using the **JESCHAR=** parameter in BBISSP00.

When IMFEXEC CMD or a TS user issues an OS/390 command with the first character equal to the command character specified by the **JESCHAR=** parameter, the command is issued by using the subsystem interface instead of SVC 34. If either the **JES3CMD=** or the **JESCHAR=** parameter is set incorrectly, JES3 commands cannot be issued by using IMFEXEC CMD or from the TS.

Command response is not available for these JES3 versions.



---

# Chapter 18 Implementing Continuous State Manager

This chapter provides information to help you implement Continuous State Manager (CSM) and attain the best possible performance.

BMC Software strongly recommends that you use the MAINVIEW Total Object Manager (TOM) application for managing started tasks. For information about this product, refer to the *MAINVIEW Total Object Manager User Guide*.

## Performance Considerations

For the best performance results, add both CSM EXEC libraries, prefix.BBPROC and prefix.BBCLIB, to the Virtual Lookaside Facility (VLF). Refer to the IBM book *TSO Customization Guide* for more information about VLF.

Define CSM EXECs (CSMUP, CSMDOWN, CSMBUILD, and CSMINIT) as high-priority MAINVIEW AutoOPERATOR EXECs in BBPARM member AAOEXP00. See Chapter 4, “Continuous Operation: Using the Dynamic Parameter Manager,” for more information about AAOEXP00 and high-priority EXECs.

# Initiating CSM

The following sections describe how to initiate CSM.

## Overriding the Default Group when CSM Is Initiated

In CSM, a group list is a collection of objects designated to run on a particular CSM partner. When you are defining objects to CSM, you must identify the group to which the object will be a member.

By default, the object belongs to the group named after the subsystem ID (SSID) of the BBI-SS PAS where CSM runs. You can override the group by specifying a different group name on the CSMALTDB parameter in the OS/390 start command for the BBI-PAS. Refer to “Subsystem (SS) JCL Parameter CSMALTDB” for more information.

However, the default group cannot be overridden when CSM is initialized for the first time. To use a group other than the default, CSM *must* first be initialized with the default group. Subsequently, a group record must be created during a CSM terminal session. Once a group record is created, it can be specified on the CSMALTDB parameter to override the default group during future initializations.

### Subsystem (SS) JCL Parameter CSMALTDB

The SS JCL contains the CSMALTDB parameter. This parameter can be used to specify the name of a CSM group that is used by CSM when it initializes, or you can specify CSMALTDB=MINIMAL, which enables you to access the CSM TS to edit objects without CSM actually controlling the tasks.

**Note:** When you modify the CSMALTDB parameter, START=COLD,VPOOL=RESET must be specified for the change to take effect.

An example of specifying the group name on the CSMALTDB parameter follows:

**CSMALTDB=SYSB**

where **SYSB** is the name of an SSID that is not local to the BBI-SS PAS that is starting CSM.

## End-of-Memory Initiated CSM EXEC (CSMEOM)

Normally, object termination can be monitored by using standard OS/390 and JES messages. However, for some situations, monitoring based on these messages is not sufficient because an object might terminate without producing the expected messages.

Use the CSM EOM (CSM End-of-Memory) EXEC to ensure that critical objects do not terminate unnoticed. The CSMEOM EXEC allows CSM to monitor object termination regardless of how the object is terminated.

CSMEOM is designed to be called from an IMFEOM EXEC with the ASID and status being passed to the EXEC. CSM detects objects that terminate at End-of-Memory by using the CSMEOM EXEC.

See the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for more information about EOM EXECs.

## Accessing CSM for the First Time

Take the following steps before you start using CSM from the terminal session:

- Step 1** Obtain and enter a valid key for the MAINVIEW AutoOPERATOR for OS/390 option in BBPARM member BBKEYS.

The MAINVIEW AutoOPERATOR for OS/390 option is required for CSM processing, which includes issuing both OS/390 commands and SYSPROG services.

- Step 2** In BBPARM member AAOPRM00, specify that the RULESET parameter includes the AAORULBC, AAORULBD, AAORULBE and AAORULBG Rule Set suffixes; for example:

```
RULESET= ( BC , BD , BE , BG )
```

These Rule Sets are shipped with the high-level qualifier *BBPARM* and are required for CSM to function.

Do not use any of the Rule Sets from your previously installed version. Use only the AAORULBC, AAORULBD, AAORULBE, and AAORULBG Rule Sets that came with this release.

In addition, CSM also requires other Rule Sets to function that must be added to this parameter. CSM requires you to create at least one group and requires each group to have a Rule Set associated with it. Once the group Rule Sets are created, add the Rule Set suffix to the RULESET parameter.

**Step 3** In BBPARM member AAOPRM00, set the RULESCAN parameter to I for INDIVIDUAL.

The RULESCAN parameter controls the global automation strategy for all MAINVIEW AutoOPERATOR Rules. For more information about Rules and automation strategy, refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

**Step 4** In BBPARM member AAOEXP00, enter the names of all CSM EXECs and ensure that these EXECs are defined as High Priority. The names of the CSM EXECs are listed in the AAOEXP00 member.

**Step 5** Create a Shared Repository data set to contain all defined CSM objects and schedules.

If the IBM library for SAA REXX/370 (program number 5695-014) is not available, customize the BBREXALT JCL found in BBSAMP to run on your system. Submit BBREXALT to install the REXX/370 alternate library. Add the BBLINK to the STEPLIB concatenation of the INITCSM batch job. If you do not run this job, the compiled REXX object in INITCSM will fail.

You can create a repository data set by one of two methods:

- Complete the step in AutoCustomization called “Allocate Shared Repository Data Set for CSM”.
- Run the batch job contained in BBSAMP member named INITCSM. For more information, refer to “Creating a Shared Repository Data Set for CSM” on page 18-7.

**Step 6** In BBPARM member BBIVAR $_{xx}$  member, specify the name of the Shared Repository.

Use the BBPARM member BBIVAR $_{xx}$  to define a preset shared variable for CSM. The SHARED variable that must be present for CSM to function properly is

$CSMREPDS = hilevel.datasetname$

where

CSMREPDS

Is the name of a shared variable that must be present for CSM to function correctly.

*hilevel.datasetname* Is the name of the Shared Repository data set created in Step 5.

**Note:** If you are implementing CSM in an environment where the MAINVIEW AutoOPERATOR subsystems have JESFLTR parameters specified in the BBISSP $xx$  member, and you are running more than one CSM system on an LPAR, you need to add a line in the BBIVAR $xx$  member:

**CSMMDSPL = 'Y'**

This action creates a shared variable that the CSM Rule Sets use to determine whether certain communications messages are suppressed or not.

**Step 7** Start the BBI-SS PAS and specify **START=COLD,VPOOL=RESET**

When the cold start is complete, the BBI Journal should contain the following message:

```
ACM003I CSM initialization complete
```

By completing these steps, CSM automatically initializes at BBI-SS PAS startup when message PM0010I is issued.

## Migrating from a Previous Version of MAINVIEW AutoOPERATOR

If you are migrating from a previous version of MAINVIEW AutoOPERATOR (either MAINVIEW AutoOPERATOR 4.1 or 5.1) and you want to migrate your database repository, you must complete the following steps to convert your database:

- Step 1** Start the BBI-SS PAS and specify **START=COLD,VPOOL=RESET**.
- Step 2** Follow the instructions in “Migrating a CSM Database from a Previous Release” on page 18-9.

### Creating a Group

- Step 1** If you are not migrating from a previous release, log on to a terminal session and access CSM. You must add a group before you can begin adding object records. Refer to *MAINVIEW AutoOPERATOR Basic Automation Guide* for more information about adding groups and objects.
- Step 2** Once you have either created a new group or converted all your MAINVIEW AutoOPERATOR 4.1 or 5.1 database repositories, enable the newly created groups with the Enable line command on the CSM Object Groups panel.

**Note:** When a converted group becomes enabled, the statistics about the objects are offloaded to the data set specified in the **Data set Name** field, as shown in Step 4 on page 18-12.

After the data is offloaded, it is available in comma-separated format. Figure 18-1 shows an example of what the contents in the data set might look like:

**Figure 18-1 Example of Offloaded Data**

```

BROWSE      BAOKMZ.KMZ1.OFFLOAD                               Line 00001813 Col 001 080
Command ==>>>                                           Scroll ==>>> PAGE
P,AAOCSM22,11:46:47.764226,20000725,SJSD,KMZ1,KMZ1
P,AAOCSM58,11:46:48.320489,20000725,SJSD,KMZ1,KMZ1
S,AAOCSM27,11:46:51.367131,20000725,SJSD,KMZ1,KMZ1
S,AAOCSM55,11:46:53.437561,20000725,SJSD,KMZ1,KMZ1
S,AAOCSM59,11:46:53.626765,20000725,SJSD,KMZ1,KMZ1
P,AAOCSM17,11:47:43.247170,20000725,SJSD,KMZ1,KMZ1
S,AAOCSM01,11:47:52.415915,20000725,SJSD,KMZ1,KMZ1
***** Bottom of Data *****
    
```

Data in a comma-separated format can be uploaded to popular spreadsheet applications, such as Microsoft Excel, where it can become input to a variety of reports.

## Creating a Shared Repository Data Set for CSM

**Note:** The information contained in this section is part of Step 5 as described in “Accessing CSM for the First Time” on page 18-3. This section contains a set of procedures that you must complete before using CSM for the first time.

Complete the procedure in this section only if you did not create a Shared Repository by completing the steps in AutoCustomization called “Allocate Shared Repository Data Set for CSM.” See the *MAINVIEW Common Customization Guide*.

Use the BBSAMP member INITCSM to create and initialize a Shared Repository data set that will contain all defined CSM objects and schedules.

To customize this job stream, follow these steps:

- Step 1** Modify the jobcard to installation standards.
- Step 2** Make the following global JCL and SYSIN changes:
  - 2.A** Change **\*HILEVEL\*** to the high-level qualifier for BMC Software product data sets.
  - 2.B** Change **\*SSID\*** to the four character subsystem (SSID) name.
  - 2.C** Change **\*VOL\*** to the volume serial number of the DASD volume on which the linear data set is to be allocated.
- Step 3** Submit the newly modified job stream.

After the job stream has run successfully, check for zero return codes for both steps. Additionally, the SYSTSPRT DD should contain output messages. The message CSM REPOSITORY SUCCESSFULLY INITIALIZED should appear at the end of this data set.

**Note:** The space allocations shown in Figure 18-2 on page 18-8 should be enough for over 2,000 objects on most device types. For more details about space allocation, refer to “Calculating Repository Size for a 3390” on page 18-8.

Figure 18-2 BBSAMP member INITCSM Sample JCL

```

//INITCSM JOB ACCOUNT ** MODIFY JOB CARD TO INSTALLATION STANDARDS **
//*-----*
//*
//* THIS JCL CREATES AND INITIALIZES THE DATA REPOSITORY USED
//* BY CSM. THE CSM REPOSITORY IS A KEYED VSAM DATASET WHICH
//* CAN BE SHARED BY ALL MEMBERS OF A CSM-PLEX.
//*
//*-----*
//*
//INITCSM PROC HILEVEL='*HILEVEL*',
//          SSID='*SSID*',
//          SYSOUT=*
//*
//DEFINE EXEC PGM=IDCAMS,REGION=3M
//SYSPRINT DD SYSOUT=&SYSOUT
//*
//INIT EXEC PGM=IRXJCL,REGION=3M,
//          PARM='INITREP'
//STEPLIB DD DISP=SHR,DSN=&HILEVEL..BBLINK
//SYSEXEC DD DISP=SHR,DSN=&HILEVEL..BBPROC
//CSMREPS DD DISP=SHR,DSN=&HILEVEL..&SSID..BBIREPS
//SYSTSPRT DD SYSOUT=&SYSOUT
//          PEND
//*
//INITIT EXEC INITCSM
//DEFINE.SYSIN DD *
DELETE (*HILEVEL*.*SSID*.BBIREPS)
SET MAXCC=0
DEFINE CLUSTER( NAME(*HILEVEL*.*SSID*.BBIREPS) -
                KEYS(128 0) -
                RECORDSIZE(14237 14237) -
                SHAREOPTIONS(4 )) -
DATA( NAME(*HILEVEL*.*SSID*.BBIREPS.DATA) -
      SPEED -
      CISZ(14336) -
      CYL(50,5) -
      VOL(*VOL*)) -
INDEX( NAME(*HILEVEL*.*SSID*.BBIREPS.INDEX) -
       TRK(10,10) -
       VOL(*VOL*))
LISTC ENT(*HILEVEL*.*SSID*.BBIREPS) ALL
/*

```

### Calculating Repository Size for a 3390

The repository has a CI size of 14,336. Each physical block of data is 7,168 bytes in length. Each track on a 3390 device can hold 7 physical blocks of data. Each record in the repository has a length of 14,237 bytes. Each cylinder on a 3390 device holds 15 tracks of data.



**Step 1** Log on to a terminal session and access the CSM Global Overview panel (as shown in Figure 18-4).

**Figure 18-4 CSM Global Overview Panel**

```

BMC Software ----- Global Overview ----- AutoOPERATOR
COMMAND ==>>                                     SCROLL ==>> CSR
                                                TGT --- AO6A

Primary Commands: EXception  Group    SORT    SAVE    ADD
                  CAleNDAR  CANCEL  UNsort  CONVERT  Build
                  Locate    CMDSHOW Xref    SAVERULE

Line Command - Enter "/" to Select Action
LC  Object  Group  System  Actual  Desired  Ind  Type
-----
___  AAOCSM07  AO6A  SJSC
___  AAOCSM08  AO6A  SJSC
___  AAOCSM09  AO6A  SJSC
___  AAOCSM10  AO6A  SJSC
___  CSBEMSTR  AO61  SJSE  STARTING  UP      XE  STC-NORM
___  MVS
***** Bottom of data *****
    
```

**Step 2** On the COMMAND line, enter the CONVERT primary command. The CSM Conversion panel is displayed (see Figure 18-5 on page 18-11).

Figure 18-5 CSM Conversion Panel

```

B ----- ERATOR
C | BMC Software ----- CSM Conversion - Input data ----- AutoOPERATOR | > CSR
  | COMMAND ==> | AO6A
P |
  | Specify the name of the data set and the member containing the CSM
  | objects to be converted into v6.1 format and stored in the
  | repository data set ESP.CAN.MV3905.AO61.BBIREPS
L |
  | Data Set Name ==> _____ | pe
  | _____ | NORM
  | Member ==> _____ | NORM
  | _____ | NORM
  | Specify the name of a data set where the Conversion Log will be
  | written. If you specify a nonexistent data set it will be allocated
  | * dynamically. | *****
  | Data Set Name ==> _____ |
  | Disposition ==> ____ (SHR/MOD) |
  | Enter END command to process and continue with the CSM Group
  | definitions or enter CANCEL to leave |
-----

```

**Step 3** Follow the instructions on the panel.

- 3.A** In the first **Data Set Name** field, enter the name of the CSM data set that you are replacing.
- 3.B** In the **Member** field, enter the member name (for example: *CSMssid*).
- 3.C** In the second **Data Set Name** field, specify the name of a data set where the conversion process logs information that is related to the conversion.

The data set may be a preallocated RECFM=FB, LRECL=80 data set or the data set may be dynamically allocated with these DCB attributes: RECFM=FB, LRECL=80, BLKSIZE=23680.

- 3.D** In the **Disposition** field, specify one of the following choices:
- MOD causes the Routine to append data to the end of the sequential data set.

- SHR causes the Routine to write data at the beginning of the data set.

The resulting Conversion Log contains information such as the conversion date and time, user ID, the input database name, the Group name, and a list of the converted objects and exceptions.

After you enter this information, the Group Detail Control panel is displayed.

**Step 4** On the Group Detail Control panel, enter information related to the CSM group that is being converted.

**Note:** You must add the Rule Set ID that you specify in the **Rule Set ID** field on this panel to the RULESET parameter in BBPARM member AAOPRM00 (refer to Step 2 on page 18-3).

Figure 18-6 shows an example of a completed Group Detail Control panel.

**Figure 18-6 Completed Group Detail Control Panel**

```

BMC Software ----- Group Detail Control - Add ----- AutoOPERATOR
COMMAND ==>                                           TGT --- KMZ1

Group Name          ==> SYSB_____
Command Prefix     ==> CSMKM_____

Rule information:
Ruleset Dataset    ==> BAKMZ.AAO61.KMZ1.UBBPARM
Ruleset Id         ==> G2          Rule Prefix ==> ACM1      Suffix ==> 0001

Object Statistics:
Dataset Name       ==> BAKMZ.KMZ1.OFFLOAD
Disposition        ==> MOD

Application information:
Author             ==> BAKMZ2
Description        ==> GROUP LIST FOR SYSB

Enter END command to process and return or CANCEL to leave
    
```

The default name of the CSM group is created from the last four characters of the CSMssid member that is being converted. You can choose your own group name by typing over the default name. If the name that you choose already exists, a pop-up window is displayed, asking for a new Group Name.

- Step 5** Enter a new Group Name in the pop-up window (Figure 18-7) or enter CANCEL to terminate the conversion.

**Figure 18-7 Pop-Up Window for New Group Name**

```
BMC Software ---- CSM Conversion - Group name ----- AutoOPERATOR
COMMAND ==>>

Group name NS61 already exists. Specify new name for CSM Group.

  Group Name ==>> _____

Enter END command to process or CANCEL to leave
```

- Step 6** As the conversion progresses, a pop-up window (Figure 18-8) is displayed and as every five objects are converted, the counter is updated.

**Figure 18-8 CSM Conversion Status Pop-Up Window**

```
BMC Software ----- CSM Conversion Status ----- AutoOPERATOR

                                CSM Conversion in Progress

                                15 of 164 objects converted
```

- Step 7** When the conversion is complete, the CSM Conversion Log is displayed (Figure 18-9 on page 18-14).

**Figure 18-9 CSM Conversion Log**

```

***** Top of Data
*****
BMC Software                CSM Conversion Log

Input database .... BAONIS.NS61.BBPARM(CSMNEW)
Converted by ..... BAONIS2 at: 16:54 on: 10/05/2000
New Group ..... NEW
Objects processed . 50
Conversion status . Successful for group NEW. No exceptions.

Objects      Exception Notes
-----      -
AAOCSMN1
AAOCSMN2
AAOCSMN3
AAOCSMN4
AAOCSMN5
AAOCSMN6
    
```

During conversion, you might encounter conflicts or mismatches with object definitions. For more information about resolving object-definition mismatches, refer to “Resolving Object-Definition Conflicts after Converting” on page 18-15.

**Step 8** Press **End** from the CSM Conversion Log.

The converted CSM objects and newly created Group are displayed on the CSM Global Overview panel (Figure 18-10 on page 18-15).

**Figure 18-10 Completed CSM Data Set Conversion**

```

BMC Software ----- Global Overview ----- Conversion Successful
COMMAND ==>>>                                     Scroll ==>> PAGE
                                                    TGT --- NS61

Primary Commands: EXception  Group      SORT      SAVE      ADD
                  CALendar   CANCEL   UNsort    CONVERT
                  Locate     CMDSHOW  Xref      SAVERULE

Line Command - Enter "/" to Select Action
LC  Object      Group-Id System   Actual      Desired      Ind  Type
-----
___  AAQAS01    NS61                UNKNOWN     UNKNOWN
___  AAONS61    NS61                UNKNOWN     UNKNOWN
___  CAS        NS61                UNKNOWN     UNKNOWN
___  CICS11     NS61                UNKNOWN     UNKNOWN
___  IMS71X     NS61                UNKNOWN     UNKNOWN
___  TSO        NS61                UNKNOWN     UNKNOWN
___  MVS

***** Bottom of data *****
    
```

**Step 9** Repeat Step 2 through Step 8 for each version 4.1 or 5.1 CSM data set and member name to be converted.

## Resolving Object-Definition Conflicts after Converting

The conversion process converts every object into one record in the Shared Repository data set. Because each object can have only one record within the Shared Repository, but be a member of more than one group, you might encounter object-definition conflicts. If a conflict is encountered, it is identified on the CSM Conversion Log data set (Figure 18-9 on page 18-14).

An object-definition conflict can occur when an object already has a record in the Shared Repository and the object is being converted as a member of a different group with different definitions (see Table 18-1 on page 18-16).

Table 18-1 on page 18-16 shows all possible definitions for an object. The first column contains the object definitions that can be different for each group. The second column contains the object definitions that must be identical for all groups containing this object.

**Table 18-1 Object Definitions**

<b>Do Not Have to Match in the Repository</b>	<b>Must Match within the Repository</b>
Parents Client Server Schedules	Object Type Restart Only Verify Force Down Start Command/EXEC User Post Start EXEC Stop Command/EXEC User Post Stop EXEC Cancel Command/EXEC Recovery Command/EXEC Start Command Time Out Stop Command Time Out Start Command Limit Start Event Text ID and Type Stop Event Text ID and Type Abnormal Termination Events TSO Send User ID Pager Contact Pager Info Alert Queue Rules Generation

You can resolve a conflict by starting CSM and by using the CSM Object Definition panel to edit each object and ensure that the object definitions for *each object in every group match exactly* (with the exception of the object’s Parents, Clients, Servers and Schedules).

If you do not want each object in every group to have identical definitions, you can also resolve a conflict by using variables in CSM object definitions that can resolve to different values for the different groups on different BBI-SS PASs. The definition can contain MAINVIEW AutoOPERATOR SHARED variables. For more information, refer to the section “Using Variables in the Object Definition Fields” in Chapter 11 of the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

Another way to resolve a conflicts is to place the new group into a different Shared Repository data set.

## Sample JCL for Copying the Shared Repository

Figure 18-11 shows an example of JCL that you can use to copy the Shared Repository data set into a new data set. You can use this sample JCL when you encounter the need to copy your Shared Repository data set into a larger data set.

Replace the data set names BAONIS.CSM.DATABASE and BAONIS.CSM.ALTDB with your data set names.

**Figure 18-11 JCL Example for Copying Shared Repository**

```
//JOB CARD
//CPY EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//F1 DD DSN=BAONIS.CSM.DATABASE,DISP=SHR <== Current Repository data set
//F2 DD DSN=BAONIS.CSM.ALTDB,DISP=SHR <== New Repository data set
//SYSIN DD *
      REPRO IFILE(F1) OFILE(F2)
/*
```



---

---

# Chapter 19 Specific MAINVIEW AutoOPERATOR Parameters and JCL

This chapter contains the following information:

- enabling REXX EXEC support
- detecting EXEC loops
- adding JES2 data to the MESSAGE ACTIVITY STATISTICS and ADDRESS SPACE application panels (OS/390 option only)

## Enabling REXX Support for EXECS

The REXX EXEC environment is disabled when MAINVIEW AutoOPERATOR is initially installed and activated. To enable REXX support, perform the following action and restart your BBI-SS PAS:

**Step 1** Add the following two additional DD statements to the BBI-SS PAS JCL

---

```
//SYSTSIN DD DUMMY  
//SYSTSPRT DD SYSOUT=s
```

---

where *s* is a SYSOUT class of your choice.

## Detecting EXEC Loops

Use the following parameters in BBPARM member AAOEXP00 to limit the CPU utilization and time that an EXEC may use. These parameters are effective only on MVS/SP2 systems and later.

Parameter	Description
TIMEXLIM=nnnn	Optional.  Specifies the CPU time limit for an EXEC. If this time is exceeded, the EXEC is terminated with a USER 3001 abend code.
PEREXLIM=nn	Optional.  Specifies the CPU usage threshold in an EXEC in percent. If an EXEC CPU usage exceeds this percentage during an interval of 15 seconds, the EXEC is terminated with a USER 3001 abend code.  <b>Note:</b> Usable CPU percentage is 100 percent. This amount includes multiprocessor machines.

Refer to Chapter 4, “Continuous Operation: Using the Dynamic Parameter Manager” for information about using the Dynamic Parameter Manager application to change these parameters. Also, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for more information about performance issues and using EXECs.

## Prioritizing and Threading of EXECs

You can modify BBPARM member AAOEXP00 so your system

- allows multithreading of EXECs and allows EXECs to run concurrently
- gives priority to certain EXECs

You can modify the following BBPARM member AAOEXP00 parameters:

<b>Keyword</b>	<b>Description</b>
EXEC=xxxxxxx	Enter the names of EXECs that you want to execute a high priority.
MAXNORM=nn	Enter the maximum number of normal priority EXECs that you want to execute concurrently. A normal priority EXEC is any EXEC not in the high priority list. Range is 1 to 99.
	<b>Note:</b> This value is limited more by BBI-SS PAS memory requirements than by the 99 MAX range.
MAXHIGH=n	Enter the maximum number of priority EXECs to execute concurrently.
	A high priority EXEC is any EXEC listed with the EXEC= keyword in AAOEXP00. Range is 1 to 99.

Refer to Chapter 4, “Continuous Operation: Using the Dynamic Parameter Manager” for information about using the Dynamic Parameter Manager application to change these parameters. In addition, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide* for more information about performance issues and using EXECs.

## Defining the Message Counter Table

MAINVIEW AutoOPERATOR monitors and counts all messages from different sources in the Rule Processor application and displays the results on the Event Activity Statistics panel.

To determine the total number of unique messages to be displayed on the message activity statistics panel, specify

**MCTSIZE=xxxx**

in BBPARM member BBISSP00.

This action defines the size of the event counter table by specifying the number of event to be maintained in storage. The value range is from 1 to 99999; the default is 510. Specify 0 to disable all unique event counting. For each unique event, 128 bytes of ECSA are used to maintain information.

The difference between the total count of events and the unique count of event on the Event Activity Statistics panel is shown in the following example.

If you want to monitor three different types of events such as

\$HASP309

\$HASP310

\$HASP311

the total number of unique events appears on the Event Activity Statistics panel in the **UNIQUE** field as

```
UNIQUE-      3
```

Each unique event can have up to 20 different associated activities; for example:

```
$HASP309  INIT   4  ACTIVE
$HASP309  INIT   2  INACTIVE
$HASP309  INIT   4  INACTIVE
```

The total number of the activities associated with the three events appears on the Event Activity Statistics panel as

```
Total messages:    60
```

See the *MAINVIEW AutoOPERATOR Basic Automation Guide* for more information about the Rule Processor application and the Event Activity Statistics panel.

## Displaying MVS JES2 Job Numbers in Address Space Application

If you have the MVS option for MAINVIEW AutoOPERATOR and you want to display JES2 job numbers and job class in the address space application, submit the JCL provided in BBSAMP member JES2MAO. This job reads the user's SYS1.HASPSRC data set to retrieve relevant JES2 offsets.

If you do not run this job, the warning message AAO229W is issued at BBI-SS PAS startup and the address space application does not show JES2 job numbers or their classes.

## Starting BBI-SS PAS before JES

If you want MAINVIEW AutoOPERATOR to automate functions that are normally performed before JES is operational, the MAINVIEW AutoOPERATOR subsystem must be started before JES. To start the BBI-SS PAS before JES:

- Step 1** Allocate a minimum of three dump data sets with RECFM=VBA and LRECL=125,BLKSIZE=1632,DSORG=PS. Make each one large enough to accommodate the largest dump you expect.
- Step 2** Add the following allocations to your BBI-SS PAS JCL, pointing to the dump data sets created in the previous step.

---

```
//BBDUMP01 DD DISP=SHR,DSN=dumpdsn1,VOL=SER=...,UNIT=...
//BBDUMP02 DD DISP=SHR,DSN=dumpdsn2,VOL=SER=...,UNIT=...
//BBDUMP03 DD DISP=SHR,DSN=dumpdsn3,VOL=SER=...,UNIT=...
```

---

BBI-SS PAS opens the first DUMP data set when the BBI-SS PAS starts. This data set closes and the next one opens after an abend occurs. When the last one has been opened and used, it will be reopened and reused over and over until dumps are reset using the BBI control command .RESET DUMP. You can issue this command from a terminal session, the OS/390 console, or an EXEC in the standard BBI command format.

These dump data sets are used when a BBI FORMATTED dump is taken that is controlled by the DUMPS= parameter in BBPARM member BBISSP00. If DUMPS=YES or DUMPS=BOTH is set in BBPARM member BBISSP00, a BBI FORMATTED dump is taken. If DUMPS=SDUMP is specified, a SVC dump is taken. These SYSUDUMP data sets are not used although they should always be allocated.

- Step 3** Catalog all data sets referenced in the SSJCL PROC in the master catalog or change all allocations to UNIT/VOLSER allocations.
- Step 4** Add a TIME= parameter to the EXEC statement in BBSAMP member SSJCL.

Code the TIME= parameter as required for your installation.

TIME=1440 is recommended.

**Step 5** Start the BBI-SS PAS with the command:

```
S SSJCL.SSID,SUB=MSTR
```

**Step 6** Change the following two DD statements in the BBI-SS PAS JCL:

---

```
//SYSTSIN DD dummy  
//SYSTSPRT DD dummy
```

---

**Note:** The terminal session (TS) can be started only after JES has initialized.

*If the MAINVIEW AutoOPERATOR SS is started before JES, you cannot use the IMFEXEC SUBMIT command nor can you use any command or program that attempts to allocate to a SYSOUT data set.*

If you set JESCNCT=YES in BBPARM member BBISSP00, you can use the IMFEXEC commands JESALLOC and JESSUBM to allocate and submit jobs.

## IMFJNUM Option and Seven-Digit JES Job Numbers (z/OS v1.2 and Later)

On z/OS v1.2 and later, JES2 and JES3 can have job numbers greater than 65,534. JES2 and JES3 can operate in either the old “five-digit” mode or the new “seven-digit” mode. With this support, you can have numbers up to 999,999 with room to expand one digit further.

For more information about JES2, refer to the \$ACTIVATE and \$TJOBDEF,RANGE=(xx,yy) commands. For more information about JES3, refer to the OPTIONS initialization statements and review the JOBNO= specification. More information is also available in the JES2 and JES3 migration manuals.

The following examples show the old five-digit format (representative, maximum, minimum):

```
JOB12345  
STC65534  
TSU00001
```

The following examples show the new seven-digit format (representative, maximum, minimum):

```
T0123456  
J0999999  
S0000001
```

You must set the AAOPRMxx option IMFJNUM=V to support job numbers greater than 99,999. The default is IMFJNUM=5. IMFJNUM=5 is the default to ensure compatibility for existing MAINVIEW AutoOPERATOR customers. The new setting will be effective after the next warm start of the BBI-PAS. BMC Software recommends that new customers specify the AAOPRMxx option IMFJNUM=V immediately.

When the IMFJNUM option is not specified or is set to five, the following statements are true:

- The variable IMFJNUM, which is available to both Rules and Rule-Initiated EXECs, is a fixed length five-digit field. The possible values are 1 to 99,999.
- IMFJNUM is set to null for events where the JES job number is greater than 99,999.
- A highlighted warning WTO (AU6020W) message is issued the first time MAINVIEW AutoOPERATOR detects a seven-digit mode job number that is between 1 and 99,999. This WTO message warns you that the seven-digit job number is incompatible with MAINVIEW AutoOPERATOR.
- A highlighted error WTO (AU6021E) message is issued the first time MAINVIEW AutoOPERATOR detects a job number greater than 99,999.

When the IMFJNUM option is set to V, the following statements are true:

- The variable IMFJNUM, which is available to both Rules and Rule-Initiated EXECs, is a variable length field without leading zeros.
- IMFJNUM=V supports all versions of JES.
- IMFJNUM=V supports both five-digit and seven-digit numbers.



---

# Chapter 20 Accessing Advanced SYSPROG Services from MAINVIEW AutoOPERATOR

MAINVIEW AutoOPERATOR is distributed with a set of basic SYSPROG commands (formerly called RESOLVE services). If you also own MAINVIEW for OS/390, you have the option to allow access to the Advanced SYSPROG services from MAINVIEW AutoOPERATOR and to execute the advanced early warning system (AEWS) in the MAINVIEW AutoOPERATOR address space.

The full set of Advanced SYSPROG services are available automatically to MAINVIEW AutoOPERATOR if you have MAINVIEW for OS/390 installed in the same SMP zones as MAINVIEW AutoOPERATOR or if you have the MAINVIEW for OS/390 data sets (both BBLINK and BBLOAD) defined in the linklist concatenation.

If neither of these conditions is met, you must make the following JCL modifications:

- Add the MAINVIEW for OS/390 BBLINK data set as the first data set on the STEPLIB DD statement in the SS JCL.
- Add the MAINVIEW for OS/390 BBLOAD data set as the first data set on the BBILOAD DD statement in the SS JCL.
- If you have combined the MAINVIEW for OS/390 BBLINK and BBLOAD data sets, make a reference to the combined data set on both STEPLIB and BBILOAD.

MAINVIEW AutoOPERATOR requires all data sets in the BBILOAD concatenation to be APF authorized. The MAINVIEW for OS/390 BBLOAD data set might not be authorized since MAINVIEW for OS/390 does not require this authorization. If the data set is not authorized, you must authorize the MAINVIEW for OS/390 BBLOAD data set before adding it to the BBILOAD concatenation.

Not all the Advanced RESOLVE commands function under MAINVIEW AutoOPERATOR. For example, the CONSOLE service does not function because it uses full screen support and the MAINVIEW AutoOPERATOR interface is based on line-mode.

The user security exit (ASTXA1UE) is invoked for each service invocation. Refer to the *MAINVIEW AutoOPERATOR Options User Guide* for more information about the user security exit.

## Executing AEWS in the MAINVIEW AutoOPERATOR Address Space

To run the advanced early warning system (AEWS) in the MAINVIEW AutoOPERATOR address space, add the LIB DD statement (as described in the *MAINVIEW Sysprog Services User Guide and Reference Manual*) to the MAINVIEW AutoOPERATOR subsystem JCL.

---

# Chapter 21 Authorization for Special MAINVIEW AutoOPERATOR Functions

Depending on how a MAINVIEW AutoOPERATOR EXEC is scheduled, it is assigned a specific type of origin. The origin is checked against a matching USERID member before default security options are checked. Additional USERID members may be needed for the following origins:

**target ID of the remote system**

for an EXEC scheduled through a message from a target

**subsystem ID of the remote BBI-SS PAS**

for an EXEC scheduled from a BBI-SS PAS message or one that is time-initiated

**jobname or RACF USERID**

for an externally initiated EXEC

## Command Authorization

Users must be authorized to issue commands or use applications against a BBI-SS PAS target system. Command authorization is defined in user authorization members of the BBPARM data set.

A summary of authorization commands for MAINVIEW AutoOPERATOR follows:

**For All MAINVIEW AutoOPERATOR Options**

<b>Command</b>	<b>Description</b>
<b>EXEC</b>	authority to schedule EXECs
<b>MVSCMD</b>	authority to issue MVS commands
<b>RULEAUTH</b>	Rules processor authorization which can be READ, UPDATE, or none

**For the MAINVIEW AutoOPERATOR for IMS Option Only**

<b>Command</b>	<b>Description</b>
<b>IMSCMD</b>	authority to issue IMS commands
<b>IMSMMSG</b>	authority to send IMS messages to LTERMs
<b>IMSTRAN</b>	authority to issue IMS transactions

**For the MAINVIEW AutoOPERATOR for OS/390 Option Only**

<b>Command</b>	<b>Description</b>
<b>RESAUTH</b>	allowed to issue authorized SYSPROG commands

For more information about issuing Advanced SYSPROG commands and the security exit, refer to the *MAINVIEW AutoOPERATOR Options User Guide*.

A description of how to use these commands is in the *Implementing Security for MAINVIEW Products*. The comment field of the USERID member of the BBPARM data set also describes the commands and their valid parameters.

---

## Chapter 22 Authorizing EXECs Scheduled by an Operator or the Rule Processor

A terminal operator on one BBI-SS PAS can schedule an EXEC on another BBI-SS PAS by using the MVS MODIFY command. In addition, a message filtered by the rules processor application can schedule an EXEC that schedules another EXEC on a different BBI-SS PAS. In both cases a BBPARM member must authorize the BBI-SS PAS reception of the EXEC. The BBPARM member name is the subsystem ID (SSID) of the BBI-SS PAS that sent the EXEC.

For example, if a terminal operator on BBI-SS PAS SS01 issues this command:

```
F SS01,%execname
```

and %execname starts an EXEC on SS02, SS02 must have a BBPARM member SS01. And if %execname issues an EXEC from SS02 to be scheduled on SS01, SS01 must have a BBPARM member SS02. MAINVIEW AutoOPERATOR sends the ORIGIN of each step in an EXEC to the receiving BBI-SS PAS to verify authorization.

If the required BBPARM authorization member does not exist, MAINVIEW AutoOPERATOR issues error message AM3119W and does not schedule the EXEC.



---

# Chapter 23 Implementing MAINVIEW AutoOPERATOR for CICS Function Security (CMRSECU)

This chapter discusses function security for MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS and how it is specified by assembling and linking the CMRSECU module using the CMRSECU macro.

This table can be reset dynamically with the command:

**.RESET PARM CMRSECU**

See the *MAINVIEW Administration Guide* for a description of the RESET control command and supported parameters.

The sample security member is distributed in BBSAMP member CMRUSECU. Use the following procedure to update this member:

- Step 1** Copy member CMRASM from **prefix.BBSAMP** to **prefix.UBBSAMP**. Make sure the new member name conforms to site naming conventions.
- Step 2** Edit the member created in Step 1. Read the comments in the member and then modify the JCL accordingly.

- Step 3** Modify the sample invocations of the CMRSECU macro according to your site's security requirements.

The CMRSECU macro can contain up to three types of statements:

- TYPE=INITIAL** The CMRSECU TYPE=INITIAL statement defines security for each
- MAINVIEW for CICS function or action service
  - MAINVIEW AutoOPERATOR for CICS action service
- TYPE=ENTRY** The CMRSECU TYPE=ENTRY statement further defines security per operator ID. (For example, a specific function may be accessed but only by a specific operator or operators.)
- TYPE=FINAL** A single CMRSECU TYPE=FINAL statement ends the input.
- Multiple sets of CMRSECU TYPE=INITIAL and CMRSECU TYPE=ENTRY statements can be specified. Each statement and its parameters are described in the sections that follow.

- Step 4** Submit the newly created JCL; check the output to insure that all steps are completed with a return code of 0.

**Note:** S106 ABENDs can occur if the newly link-edited module causes the BBLINK data set to enter secondary extents and the active BBI-SS PAS attempts to load the module. If this situation happens, stop BBI-SS PAS and restart it.

## CMRSECU TYPE=INITIAL Statement (Required)

The CMRSECU TYPE=INITIAL statement defines security for each

- MAINVIEW for CICS function or action service
- MAINVIEW AutoOPERATOR for CICS action service

To specify which target or targets the security definitions apply to, enter

**TYPE=INITIAL,TARGET=[(targetname1,targetname2,...., targetnamen. |\*)]**

**Note:** The asterisk (\*) indicates that security definitions apply to all nonspecified targets. If an \* is not specified and a nonspecified target is referenced online, the first CMRSECU TYPE=INITIAL statement is used to define function security for that target. Refer to BBSAMP member CMRUSECU for an example.

To set function security, enter YES or NO for each function parameter:

- YES indicates that the function is protected. Only operator IDs with YES specified in the CMRSECU TYPE=ENTRY statement are allowed to access that function (see “CMRSECU TYPE=ENTRY Statement by Operator ID” on page 23-7).
- NO indicates that the function is not secured. All operator IDs allowed access to MAINVIEW for CICS or MAINVIEW AutoOPERATOR for CICS are also allowed access to the function.

If the function is protected and an unauthorized operator tries to access it, an error message is issued to the operator and logged.

Display parameters apply to MAINVIEW for CICS services; if you do not have MAINVIEW for CICS installed, the parameters in the Display Parameter list do not apply. Action parameters apply to MAINVIEW for CICS functions and MAINVIEW AutoOPERATOR for CICS IMFEXEC CICS commands. For IMFEXEC CICS syntax requirements, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*.

**Note:** For MAINVIEW for CICS, functions that display information and allow actions to be taken, two function parameters are listed: one to secure access to the display; the other to secure actions taken from the display. For example, the TASK function is both a display and an action screen. Specifying TASK=NO allows unlimited access to the TASK statistics and MEMORY displays only. Specifying ALTTASK=YES secures the action services related to tasks.

for the CMRSECU TYPE=INITIAL statement, the action parameters are listed in Table 23-1 and the display parameters are listed in Table 23-2 on page 23-5.

The default for each parameter is NO.

**Table 23-1 Action Parameters for the CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements (Part 1 of 3)**

Parameter	MAINVIEW for CICS Service Description	Applicable Rule Command or IMFEXEC CICS Command
ALTAID=NO YES	specifies whether to secure the change capability for the Automatic Initiate Descriptor statistics display	N/A
ALTCLAS=NO YES	specifies whether to secure the change capability for the classes display	ALTER

**Table 23-1 Action Parameters for the CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements (Part 2 of 3)**

<b>Parameter</b>	<b>MAINVIEW for CICS Service Description</b>	<b>Applicable Rule Command or IMFEXEC CICS Command</b>
ALTCONN=NO YES	specifies whether to secure the status change capability in the CONNECT service  The expanded displays are included.	N/A
ALTDEST=NO YES	specifies the status change capability for the DEST service	DISABLE DEST ENABLE DEST PURGE DEST
ALTER=NO YES	specifies whether to secure the change capability for the virtual storage address display	ALTERVS
ALTFILE=NO YES	specifies whether to secure the status change capability in the FILE, DDIR, DSNAMES, and DATATABL services	ALLOC CLOSE FILE DISABLE FILE ENABLE FILE FREE OPEN FILE RECOVERDB STARTDB STOPDB
ALTICES=NO YES	specifies whether to secure the change capability in the ICE display	PURGE ICE
ALTPROG=NO YES	specifies whether to secure the status change capability in the PROGRAM and REMOTES services	DISABLE PROGRAM DROP ENABLE PROGRAM LOAD PROGRAM NEWCOPY PROGRAM
ALTTASK=NO YES	specifies whether to secure the task kill capability in the TASK, ENQUEUE, TERMINAL, CONSOLES, and CONNXPND services	KILL
ALTTERM=NO YES	specifies whether to secure the status change capability in the TERMINAL and CONSOLES services	ACQUIRE INSERVE OUTSERVE RELEASE SPURGE
ALTTRAN=NO YES	specifies whether to secure the status change capability in the TRAN display	ENABLE TRAN DISABLE TRAN
ALTTSUT=NO YES	specifies whether to secure the status change capability in the TSUT display	PURGE TSUT
SETTCLA=NO YES	specifies whether to secure the alteration of the attributes of a transaction class	ALTER TCLASS
SETDMPD=NO YES	specifies whether to secure the alteration of the attributes of a dump data set	ALTER DUMPDS
SETSYS=NO YES	specifies whether to secure the alteration of the CICS system attributes	ALTER SYSTEM
SETTCPS=NO YES	specifies whether to secure the alteration of a TCP/IP service	ALTER TCIPSERVICE

**Table 23-1 Action Parameters for the CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements (Part 3 of 3)**

Parameter	MAINVIEW for CICS Service Description	Applicable Rule Command or IMFEEXEC CICS Command
SETJVMP=NO YES	specifies whether to secure the alteration of the status of the JVM pool	ALTER JVMPOOL
SETDMPC=NO YES	specifies whether to secure the alteration of CICS SYSTEM and TRANSACTION DUMPCODES	ALTER DUMPCODE

**Table 23-2 Display Parameters for CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements (Part 1 of 2)**

Parameter	MAINVIEW for CICS Service Description
ABEND=NO YES	specifies whether to secure the Abend status display
AID=NO YES	specifies whether to secure the Automatic Initiate Descriptor (AID) statistics display
CLASS=NO YES	specifies whether to secure the class maximum statistics display
CONNECT=NO YES	specifies whether to secure the connection statistics display  The expanded displays are included.
DB2SYSP=NO YES	specifies whether to secure the DB2 system parameters and DBCTL displays
DB2TASK=NO YES	specifies whether to secure the DB2 task statistics and DBCTASK displays
DEST=NO YES	specifies whether to secure the destination ID information display
DLI=NO YES	specifies whether to secure the DL/1 statistics display
DUMP=NO YES	specifies whether to secure the virtual storage address display
ENQUEUE=NO YES	specifies whether to secure the enqueued resources display
EXITS=NO YES	specifies whether to secure the task-related and global user exits display
FILE=NO YES	specifies whether to secure the files statistics display in the DL/1 data base display, the file dsnames displays, and the CICS data tables displays  The expanded displays are included.
HISTORY=NO YES	specifies whether to secure the transaction history display
ICES=NO YES	specifies whether to secure the Interval Control Element (ICE) statistics display
JOURNAL=NO YES	specifies whether to secure the journal file statistics display
MONITOR=NO YES	specifies whether to secure the tuning opportunities display
NUCLEUS=NO YES	specifies whether to secure the nucleus module, LPA, and TCB displays
PLAN=NO YES	specifies whether to secure the DB2 plan display
PPST=NO YES	specifies whether to secure the DL/1 PPST/TASK activity display

**Table 23-2 Display Parameters for CMRSECU TYPE=INITIAL and TYPE=ENTRY Statements (Part 2 of 2)**

<b>Parameter</b>	<b>MAINVIEW for CICS Service Description</b>
PROBLEM=NO YES	specifies whether to secure the real-time problem analysis display  If PROBLEM=YES, you must specify OPID=xxxxxxx in the TYPE=ENTRY statement. The OPID= specification must equal the AUTOID or USRID specified in BBPARM (see page I-1) member BBIISP00.
PROGRAM=NO YES	specifies whether to secure the program status display and the remote program status display
PSB=NO YES	specifies whether to secure the DL/1 PSB statistics display
REVIEW=NO YES	specifies whether to secure the startup and current values display
SHARE=NO YES	specifies whether to secure the VSAM shared resource statistics display
SUBPOOL=NO YES	specifies whether to secure the SUBPOOL and REGION displays
SUFFIX=NO YES	specifies whether to secure the CICS module and table suffixes display
SUMMARY=NO YES	specifies whether to secure the CICS performance display
TASK=NO YES	specifies whether to secure the task statistics and MEMORY displays  The expanded displays are included.
TEMP=NO YES	specifies whether to secure the temporary storage usage statistics display  The expanded displays are included.
TERM=NO YES	specifies whether to secure the terminal statistics display in the TERMINAL and CONSOLES services  The expanded displays are included.
TIOT=NO YES	specifies whether to secure the region allocation display
TRAN=NO YES	specifies whether to secure the transaction statistics display  The expanded displays are included.
TSUT=NO YES	specifies whether to secure the TSUT statistics display
VTAM=NO YES	specifies whether to secure the VTAM buffer statistics display

## CMRSECU TYPE=ENTRY Statement by Operator ID

The CMRSECU TYPE=ENTRY statement and its parameters are used to qualify access to each

- MAINVIEW for CICS function or action service
- MAINVIEW AutoOPERATOR for CICS action service

that has been secured through the CMRSECU TYPE=INITIAL statement.

One CMRSECU TYPE=ENTRY statement is required for each operator ID that requires access to the previously secured function or service.

If the function is protected and an unauthorized operator tries to access it, an error message is issued to the operator and logged.

Specify YES or NO for each of the functions or action services that has been secured by CMRSECU TYPE=INITIAL where

- YES indicates that the operator can access the function or service that has been secured
- NO indicates that the operator cannot access the function or service that has been secured

Parameters for the CMRSECU TYPE=ENTRY statement are

OPID=xxxxxxxx|\*

OPID

Specifies an operator ID, where xxxxxxxx is

- TSO user ID for access through a terminal session
- EXCP user ID for access through an EXCP session
- VTAM user ID for access through a VTAM session

\*

Is a generic qualifier for an operator ID; for example:

AB\*

The display and action parameters are the same as the CMRSECU TYPE=INITIAL parameters.

For the CMRSECU TYPE=INITIAL statement, action parameters are listed in Table 23-1 on page 23-3 and the display parameters are listed in Table 23-2 on page 23-5.

## CMRSECU TYPE=FINAL Statement (Required)

The CMRSECU TYPE=FINAL statement has one parameter and ends this option generation; for example:

```
CMRSECU TYPE=FINAL
```

---

# Appendix A Issuing Commands from Terminal Sessions

MAINVIEW AutoOPERATOR allows OS/390, IMS, CICS, and DBCTL commands to be issued from a terminal session (TS) command line.

## Issuing OS/390 Commands

OS/390 commands may be issued from the TS either with or without response. When issued with response, the TS enters a wait and can perform no other functions while waiting for the response. If the response is received within three seconds, it is recorded in the TS user's local log and an automatic switch to this display occurs.

If the response is not received within three seconds, the short message

```
Response not available
```

is displayed on the current screen.

When commands are issued without response, the short message

```
Command issued
```

is displayed on the current screen.

If the MAINVIEW AutoOPERATOR for OS/390 (MAO) key is active in the subsystem, MAINVIEW AutoOPERATOR attempts to issue the command with response, which requires the availability of a console. If a console is not immediately available, the command is issued without response.

## Using Consoles to Issue OS/390 Commands

MAINVIEW AutoOPERATOR consoles are either subsystem consoles or Extended MCS (X-MCS) consoles. Refer to Chapter 5, “MVS Console Considerations” to determine which type of consoles are used in your environment and how they are allocated by the subsystem.

Commands issued without response are issued from a special no-response X-MCS console named SSID0000. Commands issued with response are issued using one of the Extended MCS consoles allocated to the subsystem. If a console is not immediately available, the command is issued without response from the special no-response X-MCS console named SSID0000.

The OS/390 Modify command and commands that begin with a subsystem command character are issued with response from consoles that have migration IDs (MIGIDs) associated with them. All other commands can be issued from any console. If a console is not immediately available, the command is issued without response and cannot be issued from a console that has a migration ID.

## Issuing BBI Commands

A terminal session user may record BBI commands and responses in the local log by specifying BBIRESP=LOCAL in the BBITSP00 member of the BBPROF data set.

When this option is used, the TS enters a wait and can perform no other functions while waiting for the response. If the response is received within three seconds, an automatic switch to the local log occurs.

If the response is not received within three seconds, the short message

```
Response not available
```

is displayed on the current screen.

BBI command responses are recorded in the subsystem log regardless of the user's BBIRESP= specification.

---

## Issuing IMS Commands

A terminal session user may record IMS commands and responses in the local log by specifying `IMSRESP=LOCAL` in the `BBITSP00` member of the `BBPROF` data set.

When this option is used, the TS enters a wait and can perform no other functions while waiting for the response. If the response is received within three seconds, an automatic switch to the local log occurs.

If the response is not received within three seconds, the short message

```
Response not available
```

is displayed on the current screen.

IMS command responses are not recorded in the subsystem log unless a Rule records it, regardless of the user's `IMSRESP=` specification.

## Issuing DBCTL Commands

### Issuing DBCTL Commands from a Terminal Session

When issued from a TS, the **TARGET** field must point to a DBCTL address space which is defined in `BBPARM` member `BBIJNT00` with `TYPE=IMS`. Both the command and the command response are recorded in the local BBI Journal. Only the command is logged to the BBI Journal.

To have the command response also logged to the BBI Journal, you must have `MAINVIEW AutoOPERATOR` for MVS installed. Then, create Rules to capture the messages from the MVS console. Commands can be entered in one of these two ways.

### If You Have Authority to Issue MVS Commands

- Enter a pound sign (`#`), which is the BBI MVS command prefix, followed by the MVS command recognition character, and then followed by the DBCTL command.

To have the command response also logged to the local BBI Journal, you must have `MAINVIEW AutoOPERATOR` for MVS installed.

For example, issuing the

**#!STO DATABASE ABC**

command stops the database named **ABC**.

**Note:** For this example, the ! (exclamation point) is the MVS command recognition character.

## If You Have Authority to Issue IMS Commands

- Enter the command syntax exactly as you would enter an IMS command:  
(/ **command**)

For example, issuing the

**/STO DATABASE ABC**

command stops the database named **ABC**.

---

---

# Appendix B Changing the Use of CICS Recoverable Temporary Storage

BMC Software occasionally starts a CICS transaction by using interval control services. This type of transaction must complete processing because of existing recoverable temporary storage definitions in the CICS Temporary Storage Table (TST).

If a conflict exists between the BMC Software temporary storage prefix and an existing temporary storage definition, the prefix that is used by BMC Software (CMRI) must be changed by applying the following ZAP:

```
NAME CMRINT2 CMRINT2
VER  00BC    C3D4D9C9
REP  00BC    xxxxxxxx (any nonrecoverable prefix)
```

For information about how to determine whether a conflict exists between the BMC Software temporary storage prefix and an existing temporary storage definition, refer to

- “Implementation Considerations” on page 7-2 when implementing MAINVIEW AutoOPERATOR for CICS only
- “Implementation Considerations” in the chapter “Standard Implementation Procedures” in the *MAINVIEW for CICS Customization Guide*.



---

# Appendix C CICS Transaction and Program Names

This appendix describes the transaction and program names generated by MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS.

## Transaction Entries for MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS

Table C-1 describes the transaction IDs used by MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS. Table C-2 on page C-2 lists ZAPs that may be applied to change these IDs.

**Table C-1 Transaction IDs (Part 1 of 2)**

Tran ID	Description
FCD2	performs an action request that uses CICS services for completion  It is scheduled once for each request to ensure proper serialization.
FCM1	routes messages to terminals through BMS for MAINVIEW AutoOPERATOR for CICS message broadcast
FIC2	captures and records MAINVIEW for CICS statistical data  It runs once each minute.
FST2	activates (FST2 QON), deactivates (FST2 QOFF), and controls the communications and data collection components under CICS and switches dual journals (FST2 SWITCH)  It normally runs asynchronously (without a terminal), but it can be entered from an OS/390 console defined to CICS.

**Table C-1 Transaction IDs (Part 2 of 2)**

Tran ID	Description
JNL2	<p>monitors the status of the BBI-SS and starts transaction FCD2 when CICS action requests are processed</p> <p>JNL2 is always shown as active on the TASK display, but most of that time is spent waiting (user ECB). When JNL2 starts, it remains active because it must be present in the system at all times once communication with the BBI-SS is established. JNL2 is assigned a very high internal dispatching priority.</p>
SMN2	<p>allows manual starting of communications and data collection components</p> <p>It also allows manual stopping of data collection (communications, however, remain active once started). It schedules FST2 (FST2 QON, FST2 QOFF) to perform a service and reports on the status of the components. Its use is required if the PLT is not used to start data collection and communications automatically.</p>

If your site has existing transaction IDs that are the same as those used by MAINVIEW for CICS or MAINVIEW AutoOPERATOR for CICS, or your site has standards that preclude the use of these IDs, you need to change them. To change these IDs, apply the following ZAPs:

**Table C-2 ZAPs for Transaction ID Changes**

Tran ID	ZAP Required if Transaction ID Is Changed
FCD2	NAME CMRINT2 CMRINT2 VER 00C0 C6C3C4F2 REP 00C0 xxxxxxxxx New Transaction ID
FCM1	NAME CMRINT2 CMRINT2 VER 00A8 C6C3D4F1 REP 00A8 xxxxxxxxx New Transaction ID
FIC2	NAME CMRINT2 CMRINT2 VER 00B8 C6C9C3F2 REP 00B8 xxxxxxxxx New Transaction ID
FST2	NAME CMRINT2 CMRINT2 VER 00B0 C6E2E3F2 REP 00B0 xxxxxxxxx New Transaction ID
JNL2	NAME CMRINT2 CMRINT2 VER 00B4 D1D5D3F2 REP 00B4 xxxxxxxxx New Transaction ID
SMN2	(No ZAP required)

**Note:** While the transaction IDs may change in CICS, the original transaction IDs still appear in online Help.

# Program Entries for MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS

This section provides a descriptive list of program entries generated for MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS. The lists are grouped by release-independent and release-dependent programs.

## Release-Independent Programs

The programs listed in Table C-3 can operate under any CICS release.

**Table C-3 Release-Independent Programs**

Program	Description
CMRINT2	CICS products internal table
CMRLOGM	CICS products message logging program
CSLOAD	CICS products common services program
CMRTMON	CICS products interactive startup program
CMRTMAP	CICS products interactive startup BMS mapset
CBQCMD	MAINVIEW AutoOPERATOR for CICS broadcast program
CMRXTFG	MAINVIEW for CICS verification program
OLTSCAO	start and stop MAINVIEW AutoOPERATOR for CICS
OLTVCAO	Test software compatibility

## Release-Dependent Programs

These programs operate under a specific CICS release. The release dependency is indicated by the following suffixes used with the program name. Table C-4 on page C-4 is a descriptive list of the programs.

Suffix	CICS Release
6	CICS/ESA 4.1
9	CICS Transaction Server 1.2
0	CICS Transaction Server 1.3
1	CICS Transaction Server 2.1
2	CICS Transaction Server 2.2

**Table C-4 Release-Dependent Programs (Part 1 of 2)**

<b>Program</b>	<b>Description</b>
CMRFSET6 CMRFSET9 CMRFSET0 CMRFSET1 CMRFSET2	CICS products initialization program
CMRJRN6 CMRJRN9 CMRJRN0 CMRJRN1 CMRJRN2	CICS products BBI-SS communications program
CMRFCMD6 CMRFCMD9 CMRFCMD0 CMRFCMD1 CMRFCMD2	CICS products action service program
CMRTDPX6 CMRTDPX9 CMRTDPX0 CMRTDPX1 CMRTDPX2	MAINVIEW AutoOPERATOR for CICS transient data exit program
CMRCMPX6 CMRCMPX9 CMRCMPX0 CMRCMPX1 CMRCMPX2	MAINVIEW for CICS monitoring task-related user exit program
CMRKCPX6 CMRKCPX9 CMRKCPX0 CMRKCPX1 CMRKCPX2	MAINVIEW for CICS data collection global exit program
CMRKCPX6 CMRKCPX9 CMRKCPX0 CMRKCPX1 CMRKCPX2	MAINVIEW for CICS data collection global exit program
CMRTRUE6 CMRTRUE9 CMRTRUE0 CMRTRUE1 CMRTRUE2	MAINVIEW for CICS task-related user exit program
CMRROLL6 CMRROLL9 CMRROLL0 CMRROLL1 CMRROLL2	MAINVIEW for CICS statistics output program driver

**Table C-4 Release-Dependent Programs (Part 2 of 2)**

<b>Program</b>	<b>Description</b>
CMRSLOG6 CMRSLOG9 CMRSLOG0 CMRSLOG1 CMRSLOG2	MAINVIEW for CICS statistics gathering program
CMRXEIO6 CMRXEIO9 CMRXEIO0 CMRXEIO1 CMRXEIO2	MAINVIEW for CICS task kill global user exit program
OLTSCMR6 OLTSCMR9 OLTSCMR0 OLTSCMR1 OLTSCMR2	start and stop MAINVIEW for CICS data collection
OLTVCMR6 OLTVCMR9 OLTVCMR0 OLTVCMR1 OLTVCMR2	test OLTP release compatibility (MAINVIEW for CICS)



---

---

# Appendix D Manual Activation of BBI-SS to CICS Communication

If BBI-SS to CICS communication has already been implemented, but is not started automatically through the CICS PLTPI table, it can be started manually either from an OS/390 console that has been defined to CICS, or from a CICS terminal.

For instructions on implementing BBI-SS to CICS communication, refer to

- “Implementing BBI-SS PAS to CICS Communication” in the *MAINVIEW Common Customization Guide* when implementing MAINVIEW AutoOPERATOR for CICS only
- “Standard Implementation Procedures” in the *MAINVIEW for CICS Customization Guide* (when implementing MAINVIEW for CICS or MAINVIEW AutoOPERATOR for CICS or both)

## Manual Activation from a CICS Terminal

To start BBI-SS to CICS communication from a CICS terminal:

- Step 1** Invoke the SMN2 transaction.
- Step 2** Type **YES** in the **ESTABLISH COMMUNICATIONS** field.
- Step 3** Press **Enter**.

---

## Manual Activation from an OS/390 Console that Is Defined to CICS

To start BBI-SS to CICS communication from an OS/390 console which has been defined to CICS:

Use the MODIFY command to invoke the FST2 transaction in the CICS region. For example:

**F CICSPROD, FST2**

invokes the FST2 transaction in the CICS region with the OS/390 job name of CICSPROD.

---

# Appendix E    BBPARM Data Set Members for MAINVIEW AutoOPERATOR

This section describes BBPARM data set members that you can use as samples to set up and customize MAINVIEW AutoOPERATOR.

Table E-1 describes all of the BBPARM members that are specific to MAINVIEW AutoOPERATOR. Documentation for other BBPARM members (such as BBISSP00) or that are shared among several products can be found in the *MAINVIEW Common Customization Guide*.

**Table E-1    BBPARM Data Set Members: MAINVIEW AutoOPERATOR Only (Part 1 of 3)**

<b>BBPARM Member Name</b>	<b>Description</b>
AAOALS00	<p>This member defines limits for extended storage for MAINVIEW AutoOPERATOR ALERTs and ALERT parameters on your system.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAOALT00	<p>This member is used to define the VTAM connections for sending ALERTs to PATROL EM workstations.</p> <p>This member <i>does not</i> support system variables substitution, but <i>does</i> support the LIST option on the PROCESS control card.</p>

**Table E-1 BBPARM Data Set Members: MAINVIEW AutoOPERATOR Only (Part 2 of 3)**

<b>BBPARM Member Name</b>	<b>Description</b>
AAOARP00	<p>This member is used by the MAINVIEW AutoOPERATOR Automation Reporter application to activate and control collection intervals for offline data collection.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAOEW00	<p>This member is an Elan Workstation parameter member.</p>
AAOEXP00	<p>This member defines thread values and high-priority EXEC names and limits the CPU time that EXECs can use.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAOGME00	<p>This member is used by MAINVIEW AutoOPERATOR to create General Messages Exchange (GME) connections to allow applications to communicate.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAOMQL00	<p>This member is used to specify which MQSeries queues will be made eligible for automation.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAONCD00	<p>This member is used for the MAINVIEW AutoOPERATOR ACCESS NV option.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>

**Table E-1 BBPARM Data Set Members: MAINVIEW AutoOPERATOR Only (Part 3 of 3)**

<b>BBPARM Member Name</b>	<b>Description</b>
AAOPRM00	<p>This member defines Operator Request parameters for the Operator Request application.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
AAORUL00	<p>This member contains Rule Set definitions.</p> <p>This member <i>does not</i> support system variables substitution, but <i>does</i> support the LIST option on the PROCESS control card.</p>
AAOTRN00	<p>This member contains the identification of IMS transactions that can be submitted from a MAINVIEW AutoOPERATOR EXEC or from a terminal session.</p> <p>This member <i>does not</i> support system variables substitution, but <i>does</i> support the LIST option on the PROCESS control card.</p>
AAOTSP00	<p>This member is used for the TapeSHARE for MAINVIEW AutoOPERATOR option.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>
BBIVAR00	<p>This member defines any number of preset SHARED variables and their values to be set in the SHARED pool at the COLD start of the BBI-SS PAS COLD when VPOOL=RESET is specified.</p> <p>This member supports system variables substitution and the following control card:</p> <p>PROCESS SYM=YES   NO LIST=YES   NO</p> <p>Keywords can be in any order.</p>

## System Variables Substitution in Parameter Library Members

Some MAINVIEW AutoOPERATOR BBPARM members support system variables substitution. (Refer to Table E-1 for a brief description of each BBPARM member.) With symbolic substitution, you can reduce the number of parameter libraries that you maintain. When the member is read, the value of the symbol is substituted for the symbol. The symbol must begin with an ampersand, followed by one to eight characters, and can optionally end with a period. You can define your own symbols by placing them in the IBM PARMLIB member IEASYMxx.

Symbolic names can be concatenated as a string, and names can be added at the beginning, middle or end of the string. See the following example:

```
DSN=VAM3 . &SYSPLEX . . &SYSNAME . . OFFLOAD
```

If &SYSPLEX=BMC1 and &SYSNAME=SJSD, the preceding string would convert to

```
DSN=VAM3 . BMC1 . SJSD . OFFLOAD
```

**Note:** A period at the end of a symbolic name is assumed to be a delimiter and is omitted in the conversion. If you want to generate the string BMC1SJSD, you would code &SYSPLEX&SYSNAME .

For more information about symbolic substitution, refer to “System Symbols and Symbolic Substitution” in the *IBM Initialization and Tuning Reference*.

You can control substitution on a member-by-member basis by adding a control card (a PROCESS statement) at the beginning of each member. The PROCESS statement is described in Table E-2.

**Table E-2**      **PROCESS Statement**

Control Card	Description
PROCESS SYM=YES NO LIST=YES NO	<p>If present, the PROCESS statement must be the first statement in the member and must begin in column 1.</p> <p><b>SYM=YES NO</b> This keyword indicates whether system variables substitution is allowed:</p> <ul style="list-style-type: none"> <li>• YES is the default. It indicates that symbols are replaced by their value if the calling program has enabled symbol substitution.</li> <li>• NO indicates that symbolic substitution is suppressed for this member even if the calling program has enabled substitution.</li> </ul> <p>Only the first character (Y or N) needs be specified.</p> <p><b>Note:</b> Substitution is not performed on comments or any data beyond column 71.</p> <p><b>LIST=YES NO</b> This keyword indicates whether the contents of the member being processed (all statements following the PROCESS statement) are written to the job log, one statement at a time as the member is read.</p> <p>YES causes the entire contents of the member (except the PROCESS statement) to be written to the job log.</p> <p><b>Note:</b> If substitution is performed, two WTO messages are displayed for each statement where substitution is performed. The first message contains the statement with the symbol, and the second message contains the same statement with the substituted value of the symbol. The second statement is also prefixed with &lt;&gt; to indicate that the statement is modified.</p> <p>NO is the default and indicates that the contents of this member are not written to the job log or displayed as WTO messages.</p>

You can use the Dynamic Parameter Manager (DPM) to display and alter specific parameters in members AAOEXPxx, AAOALSxx, AAOARPxx, and AAOTSPxx. For more information about DPM, see “Continuous Operation: Using the Dynamic Parameter Manager” on page 4-1.

## BBPARM Member AAOALS00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOALS00 is used by all MAINVIEW AutoOPERATOR products to define limits for extended storage for ALERTs on your system.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

For information about using the ALERTs application, refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide*; for information about using the MAINVIEW AutoOPERATOR Dynamic Parameter Manager application to update the parameters in AAOALS00, refer to “Modifying ALERT Thresholds in AAOALSxx” on page 4-7.

Table E-3 lists the parameters for AAOALS00.

**Table E-3 BBPARM Member AAOALS00 - MAINVIEW AutoOPERATOR Only (Part 1 of 3)**

Parameter	Description
MAXSTOR= <u>0</u>   nnnnn	<p>maximum amount of BBI-SS PAS private virtual storage to be allocated to ALERTs</p> <p>Storage is allocated in kilobytes. The default is 0, which means ALERTs are allocated without regard to the amount of virtual storage they occupy. BMC Software strongly recommends that you set this number to a value greater than 0.</p> <p>For example, a value of 2000 K means that you can store about 7800 ALERTs. Use the MAINVIEW AutoOPERATOR ALERT Statistics panel to estimate the average size of an ALERT on your system.</p> <p>Possible values range from 0 to 99999. A value of 0 means that no checking is done.</p>
WARNLVL1= <u>60</u>   nn	<p>first warning threshold level for queued ALERTs when the MAXSTOR parameter is set to a value greater than 0 K</p> <p>The default is 60%, which means that if MAXSTOR=1000, the warning message (AU6100W) is issued when the amount of virtual storage that is allocated to queued ALERTs reaches 600 K.</p> <p>Possible values range from 0 to 99.</p>

**Table E-3 BBPARM Member AAOALS00 - MAINVIEW AutoOPERATOR Only (Part 2 of 3)**

Parameter	Description
WARNLVL2= <u>75</u>   nn	<p>second warning threshold level for queued ALERTs when the MAXSTOR parameter is set to a value greater than 0 K</p> <p>The default is 75%, which means that if MAXSTOR=1000, the warning message (AU6100W) is issued when the amount of virtual storage allocated to queued ALERTs reaches 750 K.</p> <p>Possible values range from 0 to 99.</p> <p>Important ALERTs are not generated when ALERT storage usage reaches 100%.</p>
ALERTNV=YES   <u>NO</u>	<p>default value setting for all created ALERTs</p> <p>All ALERTs are retained across BBI-SS PAS restarts and OS/390 IPLs or they are not.</p> <p>ALERTs that are retained (or nonvolatile) are written and saved to disk.</p> <p>This parameter is equivalent to the IMFEXEC ALERT parameter RETAIN(YES   NO). Therefore, setting the ALERTNV parameter to ALERTNV=NO means that all ALERTs have a default RETAIN setting of RETAIN(NO).</p> <p>Setting the ALERTNV parameter to ALERTNV=YES mean that all ALERTs have a default RETAIN setting of RETAIN(YES).</p> <p>The default is NO.</p> <p>Regardless of which setting you specify for the ALERTNV parameter, the setting can be overridden when you create a specific ALERT, where RETAIN=YES or RETAIN=NO is explicitly set.</p> <p>Changing the ALERTNV setting takes effect only after a BBI-SS PAS warm or cold start.</p>

Table E-3 BBPARM Member AAOALS00 - MAINVIEW AutoOPERATOR Only (Part 3 of 3)

Parameter	Description
PUBLISH=ADD REPLACE NO	<p>default for the PUBLISH parameter when creating/deleting MAINVIEW AutoOPERATOR ALERTs</p> <p>Specifies whether MAINVIEW AutoOPERATOR should (by default) send ALERTs to PATROL Enterprise Manager (PATROL EM) consoles through the Generalized Message Exchange (GME).</p> <p>Possible values are as follows:</p> <p>ADD</p> <ul style="list-style-type: none"> <li>• Send ALERTs but do not override previously published ALERTs</li> </ul> <p>REPLACE</p> <ul style="list-style-type: none"> <li>• Send ALERTs but delete previously published ALERTs that had the same Key and Queue before new ALERT is published</li> </ul> <p>NO</p> <ul style="list-style-type: none"> <li>• Do not send ALERTs</li> </ul> <p>The default is ADD.</p>

## BBPARM Member AAOALT00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOALT00 is used to define the VTAM connections for sending ALERTs to PATROL EM workstations. Table E-4 on page E-9 lists the parameters that can be used to define the destination and level of diagnostics to be issued.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

**Table E-4 BBPARM Member AAOALT00 - MAINVIEW AutoOPERATOR Only**

Parameter	Description
APPLID=name	<p>Defines the value of the APPLID keyword on the ACB used by OPEN in establishing the session with VTAM</p> <p>This value must be the same name as the one entered in the VTAMLIST entry as the name of the application.</p>
DEST=device	<p>Defines a VTAM destination for the ALERTs</p> <p>This VTAM destination (a PATROL EM workstation) must be defined to the network as a 3287 printer.</p>
STATUS= <u>ON</u>   OFF	<p>(optional) Defines whether the destination is to be enabled (ON) during initialization, or left disconnected (OFF).</p> <p>The default is ON.</p>
SYNCINTV= <u>30</u>   nnn	<p>(optional) Defines the synchronization interval between MAINVIEW AutoOPERATOR and the workstation.</p> <p>At this defined interval, all current ALERTs are retransmitted to the workstation to compensate for any earlier lost transmissions or transmissions that were not processed. The default is 30 minutes.</p>
MSGLVL= <u>WARN</u>   msgtype	<p>(optional) Defines which messages are to be written to the journal; where WARN is the default and msgtype can be</p> <ul style="list-style-type: none"> <li>• NONE No messages to Journal</li> <li>• SEVERE No VTAM connection, internal errors</li> <li>• ERROR Lost or dropped connection, error limit reached</li> <li>• WARN No BBPARM member, termination request</li> <li>• INFORMATIONAL Normal start and stop messages</li> <li>• DEBUG Echo all messages in the Journal</li> </ul>

## BBPARM Member AAOARP00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOARP00 is used by the MAINVIEW AutoOPERATOR Automation Reporter application to activate and control the collection intervals for the offline data collection portion of the Automation Reporter.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

For documentation about how to use the Automation Reporter, refer to the *MAINVIEW AutoOPERATOR Basic Automation Guide*; for documentation about using the MAINVIEW AutoOPERATOR application Dynamic Parameter Manager to update the AAOARP00 member, refer to “Accessing the Dynamic Parameter Manager” on page 4-2.

Table E-5 lists the parameters for AAOARP00.

**Table E-5** BBPARM Member AAOARP00 - MAINVIEW AutoOPERATOR Only (Part 1 of 2)

Parameter	Description
DSN=(data set name)	<p>name of a user-allocated data set where collected data is offloaded</p> <p>This data set must be allocated or no data collection occurs.</p> <p><b>Data Set Requirements:</b> This user-allocated data set should be a fixed block sequential data set and the maximum record length should be 580 bytes.</p>
DISP=(SHR MOD)	<p>disposition (shared or modify) of the user-allocated data set</p> <p>The default is SHR.</p> <p>Specifying SHR causes the Automation Reporter to overwrite existing data; specifying MOD causes the collected data to be appended to the end of the data set.</p>
OFFINTVL=(60 through 1440)	<p>data offload interval in minutes</p> <p>The default is 1440 minutes (once a day).</p> <p>You can specify any value from 60 minutes (once an hour) to 1440 minutes.</p> <p>Note that the greater the frequency of offloading collected data, the faster the user-allocated data set is filled.</p>

**Table E-5 BBPARM Member AAOARP00 - MAINVIEW AutoOPERATOR Only (Part 2 of 2)**

<b>Parameter</b>	<b>Description</b>
INTVL=(0 10 through 1440)	<p>time interval (in minutes) at which data is collected</p> <p>The default is 10 minutes; specifying 0 means no data collection is performed.</p> <p>The minimum interval is 10 minutes and the maximum is 1440 (once a day).</p>
QUEUE=(MAINVIEW AutoOPERATOR ALERT queue name)	<p>one- to eight-character MAINVIEW AutoOPERATOR ALERT queue name to which any ALERTs generated by the Automation Reporter are routed</p> <p>The default queue name is ARALERT.</p> <p>The queue specified here will be created, if it does not already exist, when the first Automation Reporter ALERT is created.</p>

## BBPARM Member AAOEWD00 Parameters (MAINVIEW AutoOPERATOR Only)

Table E-6 lists the parameters for initializing BBPARM member AAOEWD00 for MAINVIEW AutoOPERATOR Elan.

**Table E-6** BBPARM Member AAOEWD00 - MAINVIEW AutoOPERATOR Only

Parameter	Description
DEBUG=YES   <u>NO</u>	generates debugging messages
HBINT= <u>30</u>   nnn	<p>defines the default heartbeat interval</p> <p>For example:</p> <p>HBINT=nn</p> <p>where nn is the number of seconds between heartbeat messages that are sent from the host SS to the Elan workstation.</p> <p>This parameter is supported for the Elan Workstation only.</p> <p>For more information, see the IMFEXEC command statements in the <i>MAINVIEW AutoOPERATOR Advanced Automation Guide</i></p>
NTINT=70   nnn	<p>defines the number of seconds for the Notify time out interval</p> <p>For example:</p> <p>NTINT=nnn</p> <p>where nnn is the number of seconds for the Notify timeout interval. The default is 70 seconds.</p> <p>This value is set once when the connection between MAINVIEW AutoOPERATOR and MAINVIEW AutoOPERATOR/Links is made.</p> <p>For more information, see the IMFEXEC command statements in the <i>MAINVIEW AutoOPERATOR Advanced Automation Guide</i></p>

## BBPARM Member AAOEXP00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOEXP00 is used by all MAINVIEW AutoOPERATOR products to define thread values, priority EXEC names, EXEC loop parameters, data set allocation, and abend count parameters.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

For information about MAINVIEW AutoOPERATOR EXECs, refer to the *MAINVIEW AutoOPERATOR Advanced Automation Guide*; for documentation about using the MAINVIEW AutoOPERATOR application Dynamic Parameter Manager to update the parameters in AAOEXP00, refer to the *MAINVIEW Common Customization Guide*.

Table E-7 lists the parameters for AAOEXP00.

**Table E-7** BBPARM Member AAOEXP00 - MAINVIEW AutoOPERATOR Only (Part 1 of 4)

Parameter	Description
AUDITMSG= <u>YES</u>   NO	specifies whether to record in the BBI-SS PAS Journal all occurrences of BLDLs against the SYSPROC data set and all user-initiated EXEC requests
EMABCNT= <u>5</u>   nn	<p>is the maximum number of abends that the EXEC Manager can sustain before it shuts down</p> <p>To restart the EXEC Manager, issue the command <b>.S COMP=EM</b>. If this command does not work, you must restart the BBI-SS PAS.</p> <p>When the EXEC Manager stops, no EXECs can run in the system. Valid values range from 1 to 99. The default is 5.</p> <p>Use this threshold to avoid recursive or persistent abends that demand attention.</p> <p>Abends in individual EXECs do not count against this threshold. The count is cumulative from the time the target BBI-SS PAS is started.</p>
EXABCNT= <u>5</u>   nn	<p>is the maximum number of abends that a single EXEC can sustain before it is disabled</p> <p>Valid values range from 1 to 99. The default is 5.</p> <p>For example, if you specify EXABCNT=10 and an EXEC abends 10 times, the EXEC is disabled after the tenth abend.</p>

Table E-7 BBPARM Member AAOEXP00 - MAINVIEW AutoOPERATOR Only (Part 2 of 4)

Parameter	Description
EXEC=name	<p>specifies the names of EXECs that are to be executed as high priority EXECs</p> <p>A high priority EXEC is any EXEC that is listed with the EXEC= keyword. This parameter can be repeated.</p>
MAXTPUT= <u>0</u>   nnnn	<p>is the maximum number of TSO TPUTs allowed for each execution of an EXEC. Valid values range from 0 to 32767</p> <p>The default is 0, which means <i>no checking is done</i>.</p> <p>TPUTs for EXECs scheduled with the WAIT=YES parameter are counted separately from the scheduling EXEC.</p>
MAXHIGH= <u>5</u>   nn	<p>is the maximum high priority EXECs to execute concurrently</p> <p>A high priority EXEC is any EXEC that is listed with the EXEC= keyword.</p>
MAXHIGHQ= <u>0</u>   nnn	<p>is the maximum number of high priority EXECs that may be queued for execution</p> <p>Valid values range from 0 to 999. The default is 0, which means <i>no checking is done</i>.</p>
MAXNORM= <u>1</u>   nn	<p>is the maximum normal priority EXECs to execute concurrently</p> <p>A normal priority EXEC is any EXEC not in the high priority list.</p>
MAXNORMQ= <u>0</u>   nnn	<p>is the maximum number of normal priority EXECs that may be queued for execution</p> <p>Valid values range from 0 to 999. The default is 0, which means <i>no checking is done</i>.</p>
OSPINUM= <u>0</u>   nnn	<p>is the maximum number of OSPI ACBs defined</p> <p>If 0, the request for a session is aborted upon encountering the first ACB that cannot be opened due to a reason other than that it is in use. Specifying the correct number of ACB definitions here helps bypass problems where ACBs have become unusable due to some other reason.</p>
OSPIPRFX= <u>OSPI</u>   name	<p>is a four-character prefix to use when selecting an OSPI ACB from the pool of available ACBs</p>
PEREXLIM= <u>00</u>   nn	<p>specifies an optional CPU percentage usage for an EXEC</p> <p>If an EXEC's CPU usage exceeds this percentage during any interval of 15 seconds after the EXEC begins, the EXEC is automatically terminated. With the default value of 0, no percentage limit is assigned.</p> <p>The CPU percentage is calculated based on the total CPU time available on 1 CPU within that 15 second interval. For example, 20% means 20% of 15 seconds. If the CPU time exceeds 3 seconds with any given 15 second interval, the EXEC will be terminated.</p>

Table E-7 BBPARM Member AAOEXP00 - MAINVIEW AutoOPERATOR Only (Part 3 of 4)

Parameter	Description
PREFIX= <u>SSID</u>   name	<p>specifies a prefix to be used when allocating a data set in an EXEC if the data set name is not enclosed within quotation marks</p> <p>This prefix also serves as the user ID associated with jobs submitted by EXECs. If the default is not specified, it will be the same as the BBI-SS PAS ID.</p>
SELLIM= <u>0</u>   nn	<p>is the maximum number of active EXECs on a single thread</p> <p>Valid values range from 0 to 99. The default is 0, which means <i>no checking is done</i>.</p> <p>This parameter is designed to control recursive calls (where EXECA schedules EXECB, that schedules EXECC, and so on) that run out of control, potentially filling up the private storage in the BBI-SS PAS and causing automation to stop.</p>
SUBXAUTH= <u>DEFER</u>   JOBNAME  RACF USER	<p><b>DEFER</b> The decision for using JOBNAME or user ID for the security check is deferred to the batch job (IMFSUBEX) execution time.</p> <p><b>JOBNAME</b> The JOBNAME is used for the IMFSUBEX security checks. This parameter overrides the origin specified on the batch job.</p> <p><b>RACF</b> The user ID is used for the IMFSUBEX security checks. This parameter overrides the origin specified on the batch job.</p> <p><b>USER</b> Same as RACF.</p>
TIMEXLIM= <u>00</u>   nnnn	<p>specifies an optional CPU time limit for an EXEC</p> <p>If the time limit is exceeded, the EXEC is automatically terminated. With the default value of 0, no time limit is assigned.</p>
TSOTIME= <u>1200</u>   nnnn	<p>specifies the time-out value (in seconds) for attached TSO commands</p> <p>A value of 0 disables timing.</p>
UCPARMS= <u>YES</u>   NO	<p>specifies whether EXEC parameters are translated to uppercase</p> <p>This parameter is important for Rule-initiated EXECs and JRNLEXEC=YES.</p>
UNITNAME= <u>SYSALLDA</u>   name	<p>specifies an esoteric unit name to be used when you allocate a data set in an EXEC and the unit parameter is omitted</p> <p>If not specified, the default shown is used.</p>

Table E-7 BBPARM Member AAOEXP00 - MAINVIEW AutoOPERATOR Only (Part 4 of 4)

Parameter	Description
WARNLVL1= <u>60</u>   nn	<p>specifies a preliminary percentage of EXECs in a queue that, when met, causes a warning message (EM6100W) to be issued</p> <p>Valid values range from 0% to 99%. The default is 60%.</p> <p>Use this threshold to set the first percentage that triggers sending a warning message.</p>
WARNLVL2= <u>75</u>   nn	<p>specifies a secondary percentage of EXECs in a queue that, when met, causes a warning message (EM6100W) to be issued. Valid values range from 0% to 99%. The default is 75%.</p> <p>Use this threshold to set a second percentage that triggers sending a warning message.</p>

## BBPARM Member AAOGME00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOGME00 is used by the MAINVIEW AutoOPERATOR General Messages Exchange (GME) to

- identify which GME nodes receive messages from and send messages to MAINVIEW AutoOPERATOR
- define the local GME node
- control which instrumentation events are traced

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

Table E-8 lists the parameters for AAOGME00.

**Table E-8 BBPARM Member AAOGME00 - MAINVIEW AutoOPERATOR Only (Part 1 of 3)**

Parameter	Description
DEBUGMSG=YES   <u>NO</u>	optional parameter that specifies whether to display GME debug messages in the BBI journal and in WTO format  Valid values are YES and NO. The default value is NO.
GMEACT=YES   <u>NO</u>	used in conjunction with the RESET PARM AAOGME <sub>n</sub> , RESYNC RECYCLE command to activate or deactivate the GME facility  Valid values are YES and NO. The default value is NO.
HOSTNAME=( <u>subsystem id of the BBI-SS PAS</u> )	name of the local GME node and is used to identify this node to other nodes  The default value is the SSID of the BBI-SS PAS. The value can consist of numeric and alphabetic characters, underscores (_), and periods (.), but the name <i>must</i> begin with an alphabetic character. The value cannot contain blanks. The maximum length is 48 characters.
LSTNPORT=nnnnn	identification of the listen port number for PATROL EM-AO functions  Client applications, such as PATROL EM AO Connection, use this port number to connect to the host node to issue <b>START</b> , <b>STOP</b> , and <b>RESTART</b> commands for PATROL EM-AO links.  Valid values are 0 through 65535 with no default value.
TGTHB=0   nnn	heartbeat interval of the remote GME in minutes  If this amount of time passes without any messages being received, the local node closes the connection.  Valid values are 0-999. The default value is 0, which disables the connection checking.
TGTIP=(network address)	the TCP/IP address or DSN host name of the machine where the PATROL EM AO Connection is running  It marks the start of a group of remote node parameters in the form "TGT*"  The address may be in IPv4, IPv6, or domain name format.
TGTMXMSL= <u>32768</u>   nnnnn	maximum message length accepted from the remote GME node  Valid values are 0-32768. The default value is 32768.

Table E-8 BBPARM Member AAOGME00 - MAINVIEW AutoOPERATOR Only (Part 2 of 3)

Parameter	Description
TGTNAME=name	<p>name of the remote GME node that is used to verify that a connection is made to the correct GME node</p> <p>Use the UNIX command <code>uname -n</code> to obtain the TGTNAME= value.</p> <p>TGTNAME= must be specified if the PATROL EM AO Connection will be issuing restart connections commands.</p> <p>The value can consist of numeric and alphabetic characters, underscores (<code>_</code>), and periods (<code>.</code>), and must contain no blanks. The maximum length is 48 characters.</p>
TGTPORT= <u>7051</u>   nnnnn	<p>listen port number that the remote GME node connects to</p> <p>TGTPORT= must match the TCP/IP port that is specified in the PATROL EM AO Connection filter path.</p> <p>Valid values are 0-65535. The default value is 7051.</p>
TGTRTC= <u>9999</u>   nnnn	<p>number of connection retries allowed if a failure occurs</p> <p>Valid values are 0-9999. A value of 0 disables the reconnection processing. A value of 9999 instructs GME to attempt to reconnect until it is successful. The default value is 9999.</p> <p>Certain platforms have a TCP/IP parameter that delays the closing of TCP/IP connections. In Solaris, this parameter is named <code>tcp_close_wait_interval</code>. GME/MVS cannot reconnect while the connection is in this <code>close_wait</code> state, because connectAAO sees it as a duplicate connection. For information regarding this parameter, refer to your platform-specific documentation.</p>
TGTRTI= <u>3</u>   nnn	<p>interval in minutes between connection retry attempts</p> <p>Valid values are 0-999. A value of 0 disables the reconnection attempt. The default value is 3.</p> <p>Certain platforms have a TCP/IP parameter that delays the closing of TCP/IP connections. In Solaris, this parameter is named <code>tcp_close_wait_interval</code>. GME/MVS cannot reconnect while the connection is in this <code>close_wait</code> state, because connectAAO sees it as a duplicate connection. For information regarding this parameter, refer to your platform-specific documentation.</p>
TRACEAPP= <u>ERROR</u>	<p>minimum severity of application trace records to create</p> <p>Valid values are NONE, ERROR, WARN, INFO, or DEBUG and are specified separately for both the local and remote GME nodes. The default value is ERROR.</p>
TRACEBUF= <u>50</u>   nnnnn	<p>amount of storage, in thousands of bytes, to allocate for the in-storage wraparound trace buffer</p> <p>The maximum value is 99999. The default value is 50.</p>

**Table E-8 BBPARM Member AAOGME00 - MAINVIEW AutoOPERATOR Only (Part 3 of 3)**

<b>Parameter</b>	<b>Description</b>
TRACEGME= <u>ERROR</u>	<p>minimum severity of GME trace records to create</p> <p>Valid values are NONE, ERROR, WARN, INFO, or DEBUG and are specified separately for both the local and remote GME nodes. The default value is ERROR.</p>
TRACELK=YES   <u>NO</u>	<p>specifies whether to perform a DNS lookup when logging IP addresses in the Trace Table</p> <p>Valid values are YES and NO. Specifying TRACELK=YES may lengthen the response times because a DNS lookup involves a network communication. The default value is NO.</p>
TRACESEC= <u>WARN</u>	<p>minimum severity of security trace records to create</p> <p>Valid values are NONE, ERROR, WARN, INFO, or DEBUG and are specified separately for both the local and remote GME nodes. The default value is WARN.</p>
TRACEZN= <u>LOCAL</u>	<p>time zone to use for time stamps in the Trace Table</p> <p>Valid values are GMT and LOCAL. The default value is LOCAL.</p>

## BBPARM Member AAOMQL00 Parameters (MAINVIEW AutoOPERATOR Only)

Use the BBPARM member AAOMQL00 parameters TYPE, QMGR, and QUEUE to specify MQSeries queues that are eligible for automation. A fourth optional parameter, OPEN, indicates how the queue should be processed. These parameters must all be typed on one line; the syntax is

TYPE ( INCL | EXCL ) QMGR ( queuemanagename ) QUEUE ( queuename ) OPEN ( S , I )

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

**Table E-9 BBPARM Member AAOMQL00 - MAINVIEW AutoOPERATOR Only (Part 1 of 3)**

Parameter	Description
TYPE   T ( <u>INCL</u>   EXCL )	<p>specifies whether <i>MAINVIEW AutoOPERATOR for MQSeries Installation and User Guide</i> should include or exclude this queue in the set of queues that are eligible for automation</p> <p>Possible values are INCL   I Specifies that MAINVIEW AutoOPERATOR should make this queue eligible for possible automation. This setting is the default.</p> <p>The abbreviation is I.</p> <p>EXCL   E Specifies that MAINVIEW AutoOPERATOR should not make this queue eligible for automation.</p> <p>The abbreviation is E.</p> <p>Abbreviations can be used; for example you can specify any of the following: T(I) or TYPE(I) or TYPE(INCL)</p>
QMGR   M(queue manager name)	<p>specifies the four-character ID for a local OS/390 queue manager that MAINVIEW AutoOPERATOR monitors</p> <p>The wildcard characters asterisk (*) and plus (+) are supported.</p> <p>This parameter is required and has no default value. <i>You must specify a queue manager name</i>, or a partial queue manager name with wildcards (a plus (+) represents one character and an asterisk (*) represents one or more characters).</p> <p>QMGR can be abbreviated to M; for example you can specify QMGR(CSQ1) or M(CSQ1)</p>

**Table E-9 BBPARM Member AAOMQL00 - MAINVIEW AutoOPERATOR Only (Part 2 of 3)**

Parameter	Description
QUEUE   U(queue name)	<p>specifies the 48-character name of the queue that MAINVIEW AutoOPERATOR makes eligible for automation</p> <p>The wildcard characters asterisk (*) and plus (+) are supported.</p> <p>This parameter is required and has no default value. <i>You must specify a queue name</i>, or a partial queue name with wildcards (a plus (+) represents one character and an asterisk (*) represents one or more characters).</p> <p>QUEUE can be abbreviated to U; for example you can specify QUEUE(SYSTEM.ADMIN.QMGR.EVENT) or U(SYSTEM.ADMIN.QMGR.EVENT)</p>

**Table E-9 BBPARM Member AAOMQL00 - MAINVIEW AutoOPERATOR Only (Part 3 of 3)**

Parameter	Description
<p>OPEN   O(EXCLUSIVE   SHARED , PROCESSOLD   IGNOREOLD)</p>	<p>specifies how a queue is opened and what happens to the messages in the queue after it is opened Possible values are as follows:</p> <p>option 1:</p> <p>EXCLUSIVE   E specifies that the queue should be opened with the MQOO_INPUT_EXCLUSIVE open option</p> <p>Other applications are unable to open the queue while MAINVIEW AutoOPERATOR has it open. It also means that if another application has the queue opened already, MAINVIEW AutoOPERATOR will be unable to open it.</p> <p>The abbreviation is E.</p> <p>SHARED   S specifies that the queue should be opened with the MQOO_INPUT_SHARED option</p> <p>This subparameter is the default for option 1. The abbreviation is S.</p> <p>option 2:</p> <p>PROCESSOLD   P specifies that all messages found on the queue when it is opened should be routed through the Rule Processor to allow automation to take place</p> <p>This option is useful for processing messages that were put on the queue while the queue was not opened by MAINVIEW AutoOPERATOR.</p> <p>The abbreviation is P.</p> <p>IGNOREOLD   I specifies that MAINVIEW AutoOPERATOR does not process the existing messages on the queue and no automation takes place for those messages</p> <p>The abbreviation is I. This subparameter is the default for option 2.</p> <p>Abbreviations can be used; for example you can specify O(E,P), which is equivalent to specifying OPEN(EXCLUSIVE,PROCESSOLD).</p> <p>Example 1: OPEN(EXCLUSIVE,PROCESSOLD) or O(E,P)</p> <p>indicates that the queue is to be opened with the MQOO_INPUT_EXCLUSIVE option, and that all existing messages found on the queue at open time should be routed through the Rule Processor for automation</p> <p>Example 2: OPEN(SHARED,IGNOREOLD) OR O(S,I)</p> <p>indicates that the queue is to be opened with the MQOO_INPUT_SHARED option, and that all existing messages found on the queue at open time should be ignored</p>

## BBPARM Member AAONCD00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAONCD00 is the Access NV BBPARM member. Table E-10 lists the parameter that is supported by AAONCD00.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

**Table E-10** BBPARM Member AAONCD00 - MAINVIEW AutoOPERATOR Only

Parameter	Description
DEBUG=YES NO	the only supported parameter  For normal operations, leave this parameter to the default setting of NO.

## BBPARM Member AAOPRM00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOPRM00 is used by all MAINVIEW AutoOPERATOR products as a general purpose member for defining Operator Request parameters in BBPARM member AAOPRM00.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

Table E-11 lists the parameters for AAOPRM00.

**Table E-11** BBPARM Member AAOPRM00 - MAINVIEW AutoOPERATOR Only (Part 1 of 3)

Parameter	Description
HONORMPF=[YES   NO]	<p>specifies whether MPF suppression is turned on (YES) or off (NO)</p> <p>HONORMPF determines whether messages are handled by the Rule Processor when the MPF suppression flag is set.</p>
MPFFILTR=(AUTO=YES   NO   token,SUP=YES   NO)	<p>specifies whether MAINVIEW AutoOPERATOR checks WTORs for the MPF settings for the AUTO and SUP keywords</p> <p><b>Note:</b> This parameter cannot be used when HONORMPF is set to HONORMPF=YES. For information about how this parameter is set and how it affects automating WTOs, refer to “Implementing Support for MPF” on page 8-1.</p>
RULESET=xx or RULESET=(xx,yy,...,zz)	<p>specifies one or more RULESETs to become active upon initialization of the BBI-SS PAS</p> <p>Format of the statement is RULESET=xx for one set, or RULESET=(xx,yy,...,zz) for multiple sets. The two-character code is appended to AAORUL.</p> <p>You also can use the 5-character suffix (where the prefix is RUL); for example, RULESET=RULxxxxx or RULESET=(RULxxxxx, RULyyyyy, RULzzzzz).</p> <p>You can specify a long list of Rule Set names by continuing to the next line. Separate each Rule Set name with a comma.</p>
ORDESC=(1,2,11)	<p>specifies the WTO descriptor codes for messages eligible for the Operator Request application</p> <p>To not check descriptor codes (or, to accept all messages), specify ORDESC=0. The default is 1,2,11.</p> <p>Messages must meet both ORROUT and ORDESC criteria to be accepted by Operator Request.</p> <p>Possible values are 0 through 16.</p>

**Table E-11 BBPARM Member AAOPRM00 - MAINVIEW AutoOPERATOR Only (Part 2 of 3)**

Parameter	Description
ORROUT=0	<p>identifies the WTO route codes for messages eligible for the Operator Request application</p> <p>The default is 0 and specifies that the OR application accept all messages regardless of the route code. A setting of 2,5 specifies that the OR application accepts messages of a 2 route code or a 5 route code.</p> <p>Messages must meet both ORROUT and ORDESC criteria to be accepted by Operator Request.</p> <p>Possible values are 0 through 128.</p>
SUPREPLY= <u>NO</u>   YES	<p>suppresses commands that are issued as replies to WTORs (write-to-operator with reply)</p> <p>Specifying YES changes the text of the command issued by a terminal session (TS) user or EXEC. For example, if an EXEC issues the command</p> <p>R 33,XYZ</p> <p>the BBI-SS PAS Journal shows the command as</p> <p>R 33,*SUPPRESSED*</p> <p>Therefore, specify YES when you want to conceal text in the BBI-SS PAS Journal (such as the use of a password). The local Journal shows the original message.</p> <p>The default is NO.</p>
MQEV=YES   <u>NO</u>	<p>specifies that <i>MAINVIEW AutoOPERATOR for MQSeries Installation and User Guide</i> should automatically enable instrumentation events for a queue manager during connection if it is not already enabled</p> <p>The default value is NO.</p>

Table E-11 BBPARM Member AAOPRM00 - MAINVIEW AutoOPERATOR Only (Part 3 of 3)

Parameter	Description
MQGINHIB=xxxxxxx   <u>JRNL</u>	<p>specifies the action MAINVIEW AutoOPERATOR should take when MAINVIEW AutoOPERATOR attempts to listen to a queue which is defined as GET(DISABLED)</p> <p>Possible settings are</p> <p><b>JRNL</b> Issue a message to the BBI Journal stating that MAINVIEW AutoOPERATOR cannot listen to the queue.</p> <p><b>WTO</b> Issue a write-to-operator (WTO) message stating that MAINVIEW AutoOPERATOR cannot listen to the queue.</p> <p><b>IGNORE</b> Take no action.</p> <p><b>ALTER</b> Alter the queue to GET(ENABLED).</p> <p>The default value is JRNL.</p>
MQNSHARE=xxxxxxx   <u>JRNL</u>	<p>specifies the action MAINVIEW AutoOPERATOR should take when MAINVIEW AutoOPERATOR attempts to listen to a queue which is defined as NOSHARE. Possible settings are</p> <p><b>JRNL</b> Issue a message to the BBI Journal stating that MAINVIEW AutoOPERATOR cannot listen to the queue.</p> <p><b>WTO</b> Issue a write-to-operator (WTO) message stating that MAINVIEW AutoOPERATOR cannot listen to the queue.</p> <p><b>IGNORE</b> Take no action.</p> <p><b>ALTER</b> Alter the queue to SHARE.</p> <p>The default value is JRNL.</p>
MQEVLPRC=xxxxxxx	<p>specifies the name of the MQSeries Event Listener PROC</p> <p>Do not use this parameter if the MQSeries Event Listener is not required. Refer to the <i>MAINVIEW AutoOPERATOR for MQSeries Installation and User Guide</i> for more information about the MQSeries Event Listener and coexistence of MAINVIEW AutoOPERATOR and other BMC Software products that require the MQSeries event queues.</p>

## BBPARM Members AAORUL00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM members AAORUL00 contain sets of distributed MAINVIEW AutoOPERATOR Rule Sets for use with MAINVIEW AutoOPERATOR Rules. These Rule Set names are therefore reserved for use to distribute MAINVIEW AutoOPERATOR Rules.

The distributed members include

Member name	Description
AAORULBA	contains a set of Rules that can be used to automate the startup and shutdown of IMS, DB2, and CICS
AAORULBB	contains a set of Rules that can be used to initialize multiple variables or functions in response to the PM0010I message
AAORULBC	contains a starter set of Rules that is used for and implemented by the MAINVIEW AutoOPERATOR application CSM
AAORULBx	contains sample Rules for the MAINVIEW AutoOPERATOR product where <i>x</i> is a predefined variable
AAORULCS	contains a starter set of Rules that can be used for the implementation of MAINVIEW AutoOPERATOR CSM
AAORULC1	contains sample Rules that intercept CICS and MAINVIEW for CICS messages
AAORULD1	contains sample Rules for DB2 solutions
AAORULD2	contains sample Rules for DB2 solutions
AAORULD3	contains sample Rules for DB2 solutions
AAORULD5	contains sample Rules for DB2 solutions
AAORULD6	contains sample Rules for DB2 solutions
AAORULD8	contains sample Rules for DB2 solutions
AAORULI1	contains sample Rules that generate ALERTs for IMS messages
AAORULJV	contains sample Rules to intercept messages and commands
AAORULM1	contains sample Rules for MAINVIEW Sample Solutions

AAORUL00 contains sample Rules for MAINVIEW AutoOPERATOR Rule Processor application

For more information about these Rule Set members, refer to the section about Rules in the *MAINVIEW AutoOPERATOR Basic Automation Guide*.

## BBPARM Member AAOTRN00 Parameters (MAINVIEW AutoOPERATOR Only)

Table E-12 lists the IMS transaction parameters for BBPARM member AAOTRN00.

**Table E-12** BBPARM Member AAOTRN00 - MAINVIEW AutoOPERATOR Only

Parameter	Description
RLTERM= <u>MASTER</u> LTERM name	IMS/AO uses the RLTERM as the input LTERM when inserting a transaction to the IMS queue for processing. The status of this LTERM should be active when an IMS transaction is submitted through the IMS/AO from a terminal session or from an EXEC. If you do not expect a response, set RLTERM to DFSMTCNT, which is always active.
TRAN=transaction name	The transaction names define a list of IMS transactions that can be submitted through the IMS/AO from a terminal session or from an EXEC.  A plus sign (+) can be used as a positional qualifier and an asterisk (*) can be used as a generic qualifier. When the plus sign and asterisk are used together, the asterisk must be the last character.

## BBPARM Member AAOTSP00 Parameters (MAINVIEW AutoOPERATOR Only)

BBPARM member AAOTSP00 is used by the TapeSHARE for MAINVIEW AutoOPERATOR component.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

For information about using the TapeSHARE for MAINVIEW AutoOPERATOR component, refer to the *MAINVIEW AutoOPERATOR Options User Guide*; for documentation about how to use the MAINVIEW AutoOPERATOR application Dynamic Parameter Manager to update the parameters in AAOTSP00, refer to the *MAINVIEW Common Customization Guide*.

Table E-13 on page E-31 lists the parameters for the MAINVIEW AutoOPERATOR TapeSHARE component.

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 1 of 7)**

Parameter	Definition
<p>ACTION=[<u>DEFAULT</u>   CANCEL   WTOR   NOHOLD   HOLD]</p>	<p>action TapeSHARE should take when Allocation Failed Event occurs because a successful GIVE cannot be performed</p> <p>Possible values and their definitions are as follows:</p> <p><b>DEFAULT</b> allows the installation's default action to occur</p> <p><b>CANCEL</b> cancels the job</p> <p><b>WTOR</b> issues a WTOR to the operator If the site does not have a defined default action, this action is the default.</p> <p><b>NOHOLD</b> enables the job to wait without holding resources</p> <p><b>HOLD</b> enables the job to wait while holding resources</p> <p><b>Example:</b></p> <p>ACTION=WTOR</p>
<p>FREE=[<u>DEALLOC</u>.   Deallocation procedure name]</p>	<p>For use with MVS SP4.x only. enables you to specify the one- to eight-character name of a deallocation procedure that starts after TapeSHARE issues the VARY OFFLINE command to GIVE a device to another image</p> <p>Scheduling the procedure triggers deallocation processing in the system GIVING the device and places the device in an OFFLINE state.</p> <p><b>Example:</b></p> <p>FREE=RELEASE</p>

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 2 of 7)**

Parameter	Definition
<p>NOGIVE=device address</p> <p>Default value is none.</p>	<p>list of device addresses (or a range of addresses) that should not be GIVEN to other partners within the TapeSHARE plex when they request devices to TAKE (in other words, a list of device addresses that are dedicated to this image)</p> <p><b>Note:</b> This parameter does not prevent you from manually GIVING a NOGIVE device.</p> <p>If a device address is not associated with this parameter, TapeSHARE assumes no devices are dedicated to this image.</p> <p><b>Example:</b></p> <p>NOGIVE=0123</p> <p>or</p> <p>NOGIVE=0120-012F</p> <p>You also can use this parameter in conjunction with the NOTAKE parameter. By specifying a device address on both these parameters, the device will be a NOGIVE-NOTAKE device and it will be excluded completely from TapeSHARE control.</p> <p>For example:</p> <p>NOGIVE=0123 NOTAKE=0123</p> <p>If the device addresses 0123 is specified on both the NOGIVE and NOTAKE parameters, the 0123 is completely excluded from TapeSHARE control. TapeSHARE does not attempt to GIVE or TAKE this device.</p>

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 3 of 7)**

Parameter	Definition
<p>NOTAKE=device address</p> <p>Default value is none.</p>	<p>list of device addresses (or a range of addresses) that this image should not TAKE when a request for resources is made</p> <p>TapeSHARE on this image will not accept (TAKE) the devices specified. If a value is not associated with this parameter, TapeSHARE assumes that it can accept any tape devices when a request for resources is made.</p> <p><b>Example:</b></p> <p>NOTAKE=0123 or NOTAKE=0120-012F</p> <p>You also can use this parameter in conjunction with the NOGIVE parameter. By specifying a device address on both these parameters, the device will be a NOGIVE-NOTAKE device and it will be excluded completely from TapeSHARE control.</p> <p>For example:</p> <p>NOGIVE=0123 NOTAKE=0123</p> <p>If the device address 0123 is specified on both the NOGIVE AND NOTAKE parameters, the 0123 device is completely excluded from TapeSHARE control. TapeSHARE does not attempt to GIVE or TAKE this device.</p>

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 4 of 7)**

Parameter	Definition
PARTNER=subsystem name	<p>BBI-SS PAS name (that can be associated with either one or a range of tape devices) that is a TapeSHARE PLEX partner</p> <p>You also can include with the BBI-SS PAS name the device address (or a range of device addresses) that TapeSHARE cannot GIVE to the named partner. (See examples below.)</p> <p>If no values are associated with this parameter, TapeSHARE assumes that any BBI node defined in BBINOD00 member is a partner within a TapeSHARE PLEX.</p> <p><b>Example 1:</b></p> <p>PARTNER=SYSA</p> <p>In this example, SYSA is a partner to this image.</p> <p><b>Example 2:</b></p> <p>PARTNER=SYSC,0123</p> <p>In this example, SYSC is a partner to this image and the device whose address is 0123 cannot be GIVEN to partner SYSC.</p> <p><b>Example 3:</b></p> <p>PARTNER=SYSB,0120-0127</p> <p>In this example, SYSB is a partner to this image and the devices whose addresses fall between 0120 and 0127 cannot be GIVEN to partner SYSB.</p>

Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 5 of 7)

Parameter	Definition
PREF=[YES   <u>NO</u>   FORCE]	<p>specifies whether Preferencing is activated</p> <p>If activated, TapeSHARE uses the devices identified on the PREFDEV parameter for only the operation (input or output) identified for that device.</p> <p>This parameter must be used in conjunction with the PREFDEV parameter.</p> <p>Specifying YES turns on Preferencing in Guideline mode. For example, in Guideline mode, if SYSA needs an output-only device but only input-only devices are available, the input-only device is GIVEN to SYSA.</p> <p>Specifying FORCE turns on Preferencing in Force mode. If Force mode is in place, the input-only device is not GIVEN to SYSA and an Allocation Failure Event occurs.</p> <p><b>Example:</b></p> <p>PREF=YES or PREF=NO or PREF=FORCE</p>
PREFDEV=device address	<p>tape device address (or a range of tape device addresses) followed by either an I for input-only or O for output-only that are identified to TapeSHARE on this image that are allocated to perform input-only or output-only functions</p> <p>Default value is none. This parameter should be used in conjunction with the PREF parameter.</p> <p>With the PREF activated, when a device is needed for an output-only operation, TapeSHARE attempts to choose from those devices identified by this parameter output-only.</p> <p>If a value is not associated with this parameter, TapeSHARE does not invoke device preferencing.</p> <p><b>Example:</b></p> <p>PREFDEV=01A0-01A3,I or PREFDEV=0130,O or PREFDEV=0150-015F,O</p>

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 6 of 7)**

Parameter	Definition
RETRYCNT=(0   2 through 10)	<p>number of times TapeSHARE attempts to satisfy a request for devices after an initial attempt has failed</p> <p>For example, suppose this system's request for devices is not satisfied and 3 retries is specified. TapeSHARE attempts three more times to satisfy this request for devices.</p> <p><b>Example:</b></p> <p>RETRYCNT=3</p> <p>This parameter is used in conjunction with the RETRYINT parameter which specifies how long TapeSHARE waits between retry attempts.</p> <p>Possible values are 0 - 10.</p>
RETRYINT=(0   30 through 300)	<p>number of seconds TapeSHARE waits for devices between attempts after an initial attempt has failed</p> <p>For example, suppose</p> <ul style="list-style-type: none"> <li>• RETRYCNT= parameter is set to 3 retries</li> <li>• RETRYINT= parameter is set to 45 seconds</li> </ul> <p>If this system's request for devices is not satisfied, TapeSHARE waits 45 seconds after the initial attempt fails and retries to satisfy the request (retry attempt 1). If this attempt fails, TapeSHARE waits another 45 seconds and retries (attempt 2). If this attempt also fails, TapeSHARE waits another 45 seconds and retries (attempt 3).</p> <p>Should all three attempts fail, an Allocation Failed Event occurs and the specified action is taken.</p> <p><b>Example:</b></p> <p>RETRYCNT=3</p> <p>Possible values are 0 - 300.</p>

**Table E-13 BBPARM Member AAOTSP00 - MAINVIEW AutoOPERATOR Only (Part 7 of 7)**

Parameter	Definition
TIMEOUT=(0   <u>120</u> through 999)	<p>amount of time (in seconds) to wait for the partners to respond to a request for devices</p> <p>When this time is reached and a successful GIVE has not been completed, an Allocation Failed Event occurs unless you have specified a number of retries on the RETRYCNT parameter.</p> <p><b>Example:</b></p> <p>TIMEOUT=20</p> <p>Possible values are 0 - 999.</p>
TRACE=[YES   <u>NO</u> ]	<p>writes diagnostic information to BBITST DD statement</p> <p><b>Example:</b></p> <p>TRACE=NO</p> <p>If YES is specified, the BBITST DD statement must appear in the BBI-SS PAS startup JCL for information to be written to it (see BBSAMP member SSJCL).</p>

## BBPARM Member BBIVAR00 Parameters (MAINVIEW AutoOPERATOR Only)

This BBPARM member defines any number of preset SHARED variables and their values to be set in the SHARED pool at the COLD start of the BBI-SS PAS COLD when VPOOL=RESET is specified.

**Note:** This member supports system variables substitution. For more information, see “System Variables Substitution in Parameter Library Members” on page E-4.

**Table E-14** BBPARM Member BBIVAR00 - MAINVIEW AutoOPERATOR Only

Parameter	Description
MYVAR999=ACTIVE  MYVAR999 = active  (Example settings)	This member contains any number of variables that are to be set at BBI-SS PAS startup when VPOOL=RESET is specified with the startup parameters. (VPOOL=RESET requires a COLD start.) These SHARED variables are set before the MAINVIEW AutoOPERATOR Rules processor is initialized. Therefore, Rules for the event type VAR will not match when these SHARED variables are initialized. However, these variables are available to be used in all other Rules.

You can create more than one BBIVARxx member (where xx is the suffix for member) and the following rules apply to determine which BBIVARxx member is processed.

- You can request multiple BBIVARxx members with the BBPARM member BBICFG.

The name of the member is constructed from the literal BBIVAR and a suffix. The default suffix is 00. An alternative suffix member can be specified through configuration of the BBI-SS PAS with a BBCFG DD statement in the PAS JCL.

- If a BBCFG DD statement is present and it contains a valid BBIVAR=xx statement, the BBIVAR=xx statement is processed.
- If no valid BBIVAR=xx statement is found in the BBCFG configuration member, a suffix of 00 is assumed.
- An example of specifying a specific BBIVARxx member in the BBCFG member is

```
BBIVAR=( 99 )
```

where 99 is the suffix for BBPARM member BBIVAR99.

To specify multiple BBIVAR<sub>xx</sub> members

**BBIVAR=(99,23,12,15)**

where **99**, **23**, **12** and **15** are suffixes for BBPARM members BBIVAR99, BBIVAR12, BBIVAR12 and BBIVAR15.

- When the BBCFG configuration member has multiple BBIVAR= statements specified, the last one is the one that the system starts with.

For example, if the BBCFG member contains

BBIVAR= ( 99 , 12 )

BBIVAR= ( 99 , 12 , 13 )

the specification BBIVAR=(99,12,13) is the one the system will start up with.

- Error or warning messages are issued for the following possible situations:
  - BBCFG contains a valid BBIVAR=xx statement but the member was not found.
  - BBCFG contains syntax errors. Syntax errors in the configuration member might cause the default BBIVAR00 to be processed.
  - The BBIVAR<sub>xx</sub> member contains syntax errors.

The following syntax rules apply to processing within the BBIVAR<sub>xx</sub> member:

Any number of entries can be specified.

- Each line is restricted to 72 characters and entries cannot be continued over multiple lines.
- Variables names can have a maximum length of 32 characters.
- An asterisk in column one indicates a comment line and causes the line to be ignored.

- An assignment statement can be in either of the following formats:

```
VARIABLE = 'VALUE'
```

```
VARIABLE = value
```

The first format must be used if the variable value contains blanks; otherwise, blank spaces are ignored. For example:

```
MYVAR = A B C      - MYVAR is set to A
```

```
MYVAR = 'A B C'   - MYVAR is set to A B C
```

- Variable names are always translated to uppercase letters.
- Variable names must conform to TSO/E REXX syntax.
- Variables are created in the variable pool exactly as specified. REXX stem variables do not experience any form of substitution. For example, the following specification creates a variable called A and a variable called I.A, not I.0:

```
A = 0
```

```
I.A = 5
```

- Variables beginning with the letter Q cannot be assigned. Q-variables are BBI-SS PAS internal and have read-only access.
- If the variable itself is to contain a single quotation mark ('), enclose the string in double quotation marks ("). If the variable is to contain double quotation marks, enclose the string in single quotation marks (').

If any of the preceding conditions are not met, an error or warning message is issued, accompanied by an additional message that identifies the invalid line. Invalid entries are skipped, and processing continues.

Examples of valid entries:

```
TEST1 = 25
```

```
test2 = 15
```

```
end_of_day = TRUE
```

```
COMMENT = 'SHIFT CYCLE'
```

```
TEST1.A = 5
```

Examples of invalid entries:

'TEST1' = 25

TEST1\* = 255

COMMENT = 'SHIFT CYCLE (End quotation mark is missing.)

QSMID = SYSB (Q-variables are reserved and read-only.)



---

# Appendix F Sample Data Set Members

The BBSAMP data set contains sample members that you can edit for your site's use.

## Sample Members for MAINVIEW AutoOPERATOR Only

Table F-1 describes BBSAMP members used only for MAINVIEW AutoOPERATOR.

**Table F-1 BBSAMP Data Set Members for MAINVIEW AutoOPERATOR (Part 1 of 2)**

<b>BBSAMP Member Name</b>	<b>Description</b>
\$EXECJOB	indicates the EXEC testing batch job
\$EXECTST	indicates the EXEC testing CLIST
\$TESTDAT	indicates the EXEC testing sample test case
AOALRTDF	defines resources to RACF for MAINVIEW AutoOPERATOR ALERT queue name resources that control a TS-user's authority to delete ALERTs and ALERT queues through the ALERT Detail and Stats panels
AOALRTPE	permits users to MAINVIEW AutoOPERATOR ALERT queue name resources that control a TS-user's authority to delete ALERTs and ALERT queues through the ALERT Detail and Stats panels
AOAPPLDF	defines all MAINVIEW AutoOPERATOR application resources to RACF
AOAPPLPE	permits users to MAINVIEW AutoOPERATOR application resources
AOCMDDF	defines all MAINVIEW AutoOPERATOR command-level resources to RACF
AOCMDPE	permits users to MAINVIEW AutoOPERATOR command-level resources

Table F-1 BBSAMP Data Set Members for MAINVIEW AutoOPERATOR (Part 2 of 2)

BBSAMP Member Name	Description
AOEXECDF	defines all MAINVIEW AutoOPERATOR EXEC name resources to RACF
AOEXECPE	permits users to MAINVIEW AutoOPERATOR EXEC name resources
AOANYDF	defines all MAINVIEW AutoOPERATOR AOAnywhere functions to RACF
AOANYPE	permits users to MAINVIEW AutoOPERATOR AOAnywhere functions
AOPARMDF	defines all MAINVIEW AutoOPERATOR DPM member name resources that control a TS-user's authority to update, read, add, or activate a DPM member through the Dynamic Parameter Manager application
AOAPPLPE	permits users to all MAINVIEW AutoOPERATOR DPM member name resources that control a TS-user's authority to update, read, add, or activate a DPM member through the Dynamic Parameter Manager application
CAOTTAB	contains a macro used to generate the list of terminal types supported by the MAINVIEW AutoOPERATOR for CICS Broadcast application
CICSTART	contains a sample EXEC used to automate the start up of the CICS environment
CICSTERM	contains a sample EXEC used to automate the shut down of the CICS environment.
DB2START	contains a sample EXEC used to automate the start up of the DB2 environment
DB2TERM	contains a sample EXEC used to automate the shut down of the DB2 environment
DIVDEF	creates the linear data set used by MAINVIEW AutoOPERATOR to store a variety of data, including nonvolatile ALERTs and data recorded by the Automation Reporter  The subsystem <i>must</i> be down prior to submitting this job stream.
DIVUTIL	copies the BBIDIV data set to a newly allocated data set or produces a report of space utilization in the BBIDIV
IMSSTART	contains a sample EXEC used to automate the start up of the IMS environment
IMSTERM	contains a sample EXEC used to automate the shut down of the IMS environment
DTABJCL	contains sample JCL to assemble and link CAODTAB
JES2MAO	contains sample JCL to assemble with the user version of SYS1.HASPSRC
NAICMDS	defines NAIEXEC and NAISTUB to NetView
NAIINIT	initializes an Access NV controlled OST
NAIOPT00	defines the BBI-SS PAS that communicates with NetView and OSTs used by that BBI-SS PAS
NAIPROF	contains a sample OST initialization profile
NAISVAR	sets global variables for Access NV tasks
NAITASK	defines NATASK as a NetView optional subtask (OPT)
RVARTST	contains an EXEC that shows settings of variables
TTABJCL	contains sample JCL to assemble and link CAOTTAB

# Sample Members for MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS

Table F-2 describes BBSAMP members used by both MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS.

**Table F-2 BBSAMP Data Set Members for MAINVIEW AutoOPERATOR for CICS and MAINVIEW for CICS**

<b>BBSAMP Member Name</b>	<b>Description</b>										
CMRASM	<p>sample JCL to assemble and link-edit any of the following modules:</p> <p><b>Note:</b> Descriptions for each of these modules can be found in this appendix.</p> <table> <thead> <tr> <th><b>Name</b></th> <th><b>Product</b></th> </tr> </thead> <tbody> <tr> <td>CMRPRBT</td> <td>MAINVIEW for CICS</td> </tr> <tr> <td>CMRRAPM</td> <td>MAINVIEW for CICS</td> </tr> <tr> <td>CMRSECU</td> <td>MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS</td> </tr> <tr> <td>CMRSOPT</td> <td>MAINVIEW for CICS</td> </tr> </tbody> </table>	<b>Name</b>	<b>Product</b>	CMRPRBT	MAINVIEW for CICS	CMRRAPM	MAINVIEW for CICS	CMRSECU	MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS	CMRSOPT	MAINVIEW for CICS
<b>Name</b>	<b>Product</b>										
CMRPRBT	MAINVIEW for CICS										
CMRRAPM	MAINVIEW for CICS										
CMRSECU	MAINVIEW for CICS and MAINVIEW AutoOPERATOR for CICS										
CMRSOPT	MAINVIEW for CICS										
CMRCSDES	sample JCL containing RDO definitions for CICS/ESA and CICS Transaction Server regions										
CMRPLT	macro used to generate valid PLT entries for CICS/ESA and CICS Transaction Server regions										
CMRSECU	security for CICS option of MAINVIEW AutoOPERATOR and MAINVIEW for CICS										
CMRUSECU	sample for defining function security										

# Sample Members for MAINVIEW AutoOPERATOR for IMS, MAINVIEW for IMS, and MAINVIEW for DBCTL

Table F-3 describes BBSAMP members used only for MAINVIEW AutoOPERATOR for IMS, MAINVIEW for IMS, and MAINVIEW for DBCTL.

**Table F-3 BBSAMP Data Set Members for MAINVIEW AutoOPERATOR for IMS, MAINVIEW for IMS, and MAINVIEW for DBCTL**

<b>BBSAMP Member Name</b>	<b>Description</b>
ICOPY7	sample installation JCL that is used to copy BMC Software modules to IMS.RESLIB data set available to the IMS 4.1 control region
ICOPY8	sample installation JCL that is used to copy BMC Software modules to IMS.RESLIB data set available to the IMS 5.1 control region
ICOPY9	sample installation JCL that is used to copy BMC Software modules to IMS.RESLIB data set available to the IMS 6.1 control region
IMRUVTF	interface from MAINVIEW for IMS/AO to the BMC Software Delta IMS virtual terminal  If this module is present, MAINVIEW for IMS/AO can access all Delta IMS lines, nodes, and LTERMs (CLB, CTB, and CNT).
LINKNUC	sample JCL that is used to link BMC Software AOI exit IBAOUE3 into the IMS nucleus (IMS version 4.1 and earlier)

---

---

# Appendix G Expanding the PROFILE Pool Data Set

This appendix discusses the XTLOAD utility that is located in BBSAMP member XTJCL. You use this utility to expand the PROFILE pool data set.

The PROFILE variables are stored in the PROFILE pool data set. When this data set becomes full, you must allocate and initialize a new, larger data set.

You can use the XTLOAD utility to copy all the PROFILE variables from the old data set into the new data set. You can identify a full PROFILE pool data set if requests to the data set from IMFEXEC VPUT or IMFEXEC VPUTL fail with return codes of 20 or 8 respectively.

**Note:** Do not use IEBGENER to copy the smaller data set into the new, larger data set. You must use the XTLOAD utility for MAINVIEW AutoOPERATOR to recognize the additional space.

## Using the XTLOAD Utility

To expand the PROFILE pool data set, use the XTLOAD utility. You must bring down MAINVIEW AutoOPERATOR before you can use this utility.

XTLOAD copies the content of the existing BBIVARS data set into a larger data set. After successful execution of XTLOAD, the new data set can be specified on the BBIVARS DD statement. Alternatively, the new data set can be renamed to match the original data set name on the BBIVARS DD statement.

---

Use the following sample JCL to execute XTLOAD.

**Figure G-1 Sample JCL to Execute XTLOAD**

---

```
//STEP1 EXEC PGM=XTLOAD
//STEPLIB DD DSN=hilevel.???? .BBLINK,DISP=SHR
//BBIVARS DD DSN=old.bbivars.dataset,DISP=OLD
//NEWVARS DD DSN=new.bbivars.dataset,
//      DISP=(NEW,CATLG),UNIT=SYSDA,
//      SPACE=(TRK,(nn))
//SYSPRINT DD SYSOUT=R
```

---

Modify the sample JCL as follows:

- Step 1** Provide a valid job card.
- Step 2** On the STEPLIB DD statement, specify the BBLINK load library containing XTLOAD.
- Step 3** On the BBIVARS DD statement, specify the name of the existing PROFILE pool data set.
- Step 4** On the NEWVARS DD statement, specify the name of the new PROFILE pool data set.

The NEWVARS data set must be larger than the BBIVARS data set; otherwise, XTLOAD will terminate and write an appropriate error message to the SYSPRINT DD.

The XTLOAD return codes are

- 4 NEWVARS, BBIVARS, or SYSPRINT DD statement is missing.
- 8 NEWVARS DD data set is not larger than the BBIVARS DD data set.

If the BBIVARS data set is full, the IMFEXEC VPUTL command receives a return code of

- 8 BBIVARS data set is full.

---

---

# Appendix H AutoRULE Worksheet

This worksheet has five sections that you must fill in before you attempt to complete the AutoCustomization step, “Run AutoRULE for AutoOPERATOR, Event Management Utility.” Each section corresponds to a panel in AutoCustomization. “Implementing AutoRULE” on page 15-1 provides documentation that describes the AutoCustomization steps.

## Section 1: MPF List Conversion Parameters

Enter the information about converting a MPFLST $xx$  member to MAINVIEW AutoOPERATOR Rules.

MPF Data Set Name \_\_\_\_\_

MPF Suffix \_\_\_\_\_

Two-character suffix of the SYS1.PARMLIB member MPFLST $xx$

Fill in the information in the following table so that AutoRULE can convert all USEREXIT references to AutoRULE records (which are later converted to MAINVIEW AutoOPERATOR Rules). For each of the USEREXITs that you want to convert, you can specify one of the following flags:

- SAVE (to indicate that AutoRULE should not convert the MPF record)
- IGNORE (to indicate that AutoRULE should ignore MPF records with that USEREXIT)
- A flag from “Section 2: AutoMate Conversion Parameters” on page H-3.

USEREXIT Name	Action Flag

Specify a default action for USEREXITs not explicitly listed in this table.

---



List the Started Tasks to exclude from abend notification:

---

---

---

List the user IDs to exclude from abend notification:

---

---

---

## Section 4: Creating Rules for Production Job Abends

Use this section of the worksheet to help create a Rule that generates an ALERT when a production job abends.

Fields	User-Specified Value
Job Name	_____
User ID	_____
RACF Group	_____
Job Class	_____
Accounting Information	_____

## Section 5: Creating Rules for Production Job Abends

Use this section of the worksheet to tell AutoRule what to do with ALERTs and other notification. AutoRULE sends notifications to the user IDs you specify. If you specify more than one TSO ID, separate them with spaces.

If There Are Problems in This Area	Send a Notification to This User ID
OS/390	_____
CICS	_____
IMS	_____
DB2	_____
Networking	_____
DASD	_____
Unauthorized Commands	_____
Automation	_____

Use this table to list the names of ALERT queues to which you want AutoRULE to send ALERTs. To specify more than one queue name, separate the queue names with two colons (: :).

If There Are Problems in This Area	Send ALERTs to This ALERT Queue Name
OS/390	_____
CICS	_____
IMS	_____
DB2	_____
Networking	_____
Tape	_____
Batch	_____
All other ALERTs	_____

List any additional parameters here:

- ALERT Target name

\_\_\_\_\_

- Default Beeper Name

\_\_\_\_\_

- Default Beeper Information

\_\_\_\_\_

# Describing AutoRULE Flags

This section describes the flag characters that AutoRULE uses while creating MAINVIEW AutoOPERATOR Rules. These flags are used during conversion of MPF list automation or AutoMate Rules to MAINVIEW AutoOPERATOR Rules.

You can manually edit the KnowledgeBase to include or exclude a flag. The flag must be in lowercase characters and start in column 1. The types of flags are

- Selection: tells AutoRULE what to use as the Rule's selection criteria
- Action: tells AutoRULE what to use as the Rule's action specifications
- Special: used by AutoRULE for comments and additional data

## Selection Flags

The following list indicates the flags for a Rule's selection criteria.

<b>p</b>	production jobs only
<b>t</b>	Started Tasks only
<b>j</b>	Batch jobs only
<b>u</b>	TSO users only
<b>g</b>	Message text (including message ID)
<b>h</b>	WHEN clause on the #h record

## Action Flags

The following list indicates the flags for a Rule's actions:

<b>a</b>	Alert
<b>b</b>	page a person
<b>c</b>	command is on #c record
<b>e</b>	EXEC name is on #e record
<b>l</b>	log message
<b>n</b>	notify a person
<b>r</b>	reply is on #r record
<b>s</b>	suppress
<b>v</b>	Set variable; specified on the #r record
<b>w</b>	ALERT/REWORD; text is on the #w record
<b>x</b>	reject a command
<b>z</b>	delete an ALERT; ALERT message ID in on the #z record

## Special Flags

The following list indicates special flags and their descriptions:

.	comment
*	comment
>	comment
<b>d</b>	disabled Rule
%	event type from here on in current member
<b>#x</b>	for g,h,b,c,e,r,v,w, and y: additional information required by the flag
<b>r</b>	reply is on #r record
<b>s</b>	suppress
<b>v</b>	set variable; specified on the #r record
<b>w</b>	ALERT/REWORD; text is on the #w record
<b>x</b>	reject a command
<b>z</b>	delete an ALERT; ALERT message ID in on the #z record



---

---

# Appendix I      How Product Libraries Should Be Used

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 release 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 libraries are created during product installation. The installation procedures are described in the *OS/390 and z/OS Installer Guide*.

- site-customized product libraries

These libraries are created for you by AutoCustomization, or you can create them manually. The AutoCustomization procedures are described in the *OS/390 and z/OS Installer Guide*. Manual procedures are described in Implementation Guide for BBI-2 products.

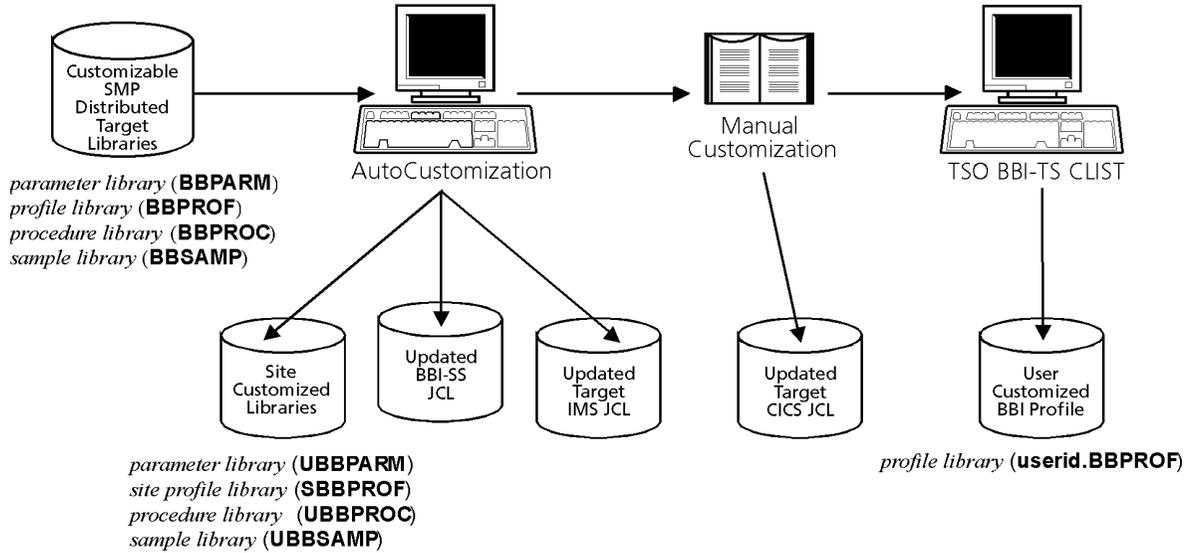
- product-user libraries

A user profile (userid.BBPROF) is created during terminal session initialization if one does not exist already.

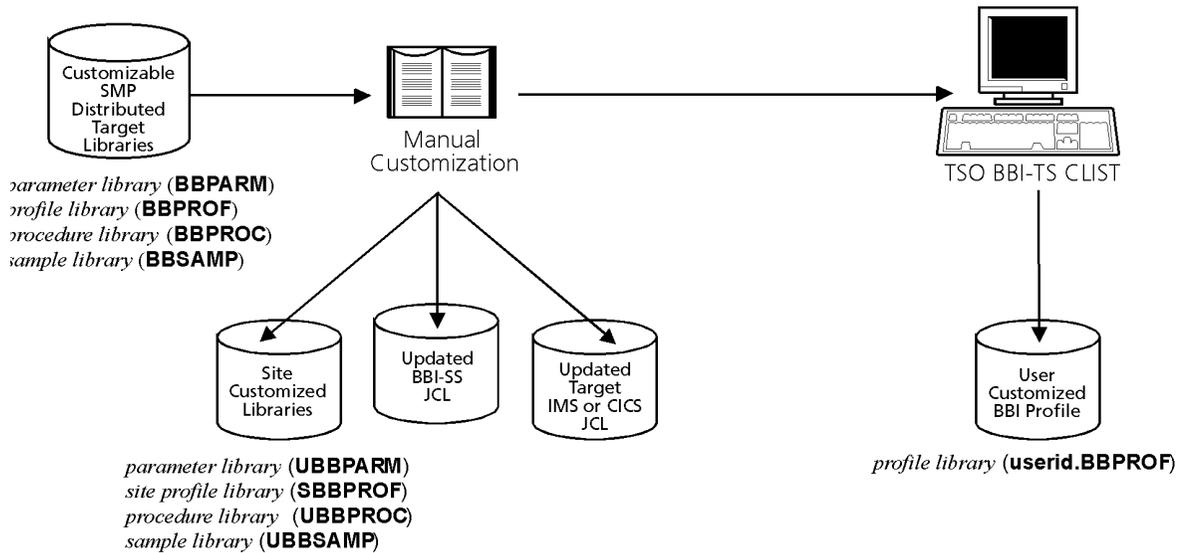
Figure I-1 on page I-2 shows how the product libraries should be configured.

Figure I-1 BBI-2 Data Set Customization

## AutoCustomization



## Manual Customization



The distributed libraries consist of the following libraries:

Parameter library	Product parameters  Members in this library contain BBI-2 parameter definitions 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, PF key, and cycle refresh definitions for a terminal session
Procedure library	MAINVIEW AutoOPERATOR execute command lists (EXECS)  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.

## Parameter Library

Each member in this data set contains parameters for a separate product or BBI-2 support component. Your site can have several parameter libraries, a distributed version and one or more site-customized versions. A BBIPARM DD statement can concatenate multiple parameter library data sets. With this technique, a site parameter library set can be created and a data set individualized for each BBI-SS 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. Use this copy to make any parameter library changes described in this manual.

### AutoCustomization

If you used AutoCustomization successfully 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, follow these steps:

- Step 1** Create your own UBBPARM data set.
- Step 2** Copy the members you need from the distributed BBPARM library into the UBBPARM data set you created.
- Step 3** Change the copied members for your needs.

## IMS-Specific Parameter Library (IMF and MAINVIEW AutoOPERATOR for IMS Only)

A parameter library, `ibbparm`, where `ibbparm` represents a user-defined parameter library, may be created for each IMS so that each has its own `IMFSYS00` member plus any other members that you want to make unique to an IMS target. `IMFSYS00` contains `BBI-SS` to IMS communication parameters. The `SUBSYS` communication parameter establishes communication between IMS and the `BBI-SS` for the IMF and MAINVIEW AutoOPERATOR for IMS products.

### Manual Customization

A parameter library for a specific IMS must be created manually and copied from the distributed parameter library as described in Customization Guide for MAINVIEW for IMS products. Use this copy of the parameter library to make product changes that are to be unique to a specific IMS target. For example, different Event Collector initialization parameters can be defined for a specific IMS by copying the `IMFECPO0` member from the distributed parameter library to your user-defined parameter library.

# Sample Library

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 all the BBI-2 products 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 successfully 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.

You can use UBBSAMP to modify other BBI-2 product samples. If you need to modify a product-specific sample as described in this manual, follow these steps:

- Step 1** Verify that the member was not copied by AutoCustomization to UBBSAMP.
- Step 2** Copy the sample member you need from the distributed BBSAMP to UBBSAMP.
- Step 3** Make the change in the copied member.

### Manual Customization

If you are customizing the product manually, follow these steps:

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

## Profile Library

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 for a BBI-TS. Other members are dynamically created by BBI applications. 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 terminal session (BBI-TS) 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
- 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 successfully when you installed your product libraries as described in the *OS/390 and z/OS Installer Guide*, it created a SBBPROF data set for you and customized its members.

### Manual Customization

If you are customizing the product manually, follow these steps:

- Step 1** Create a common SBBPROF data set for your site.
- Step 2** Copy the applicable member or members that you need from the distributed BBPROF library into the SBBPROF data set you created.
- Step 3** Change the member or 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 terminal session 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 Library (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 for MAINVIEW AutoOPERATOR, or you can create it manually.

## AutoCustomization

If you used AutoCustomization successfully 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, follow these steps:

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

### Manual Customization

If you are customizing MAINVIEW AutoOPERATOR manually, follow these steps:

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

## BBI-2 Product Library Use Summary

Table I-1 summarizes how the BBI-2 product libraries should be configured.

**Table I-1 BBI-2 Product Library Configuration (Part 1 of 2)**

<b>Distributed Library</b>	<b>Customized Library</b>	<b>Created by AutoCustomization?</b>	<b>Allocated to Which Address Space?</b>	<b>Allocated to What DD Name?</b>
BBSAMP	UBBSAMP	Yes, AutoCustomization creates UBBSAMP.	None	None
BBSAMP	UBBPARM	Yes, AutoCustomization creates UBBPARM.	BBI-SS CICS target IMS target	BBIPARM
	ibbparm	No, ibbparm is a user-defined library that must be created manually.	IMS target	IMFPARM
BBPROC	UBBPROC	Yes, AutoCustomization creates UBBPROC but only for MAINVIEW AutoOPERATOR.	BBI-SS	SYSPROC

**Table I-1 BBI-2 Product Library Configuration (Part 2 of 2)**

<b>Distributed Library</b>	<b>Customized Library</b>	<b>Created by AutoCustomization?</b>	<b>Allocated to Which Address Space?</b>	<b>Allocated to What DD Name?</b>
BBPROF	userid.BBPROF or SBBPROF or both	userid.BBPROF: No, AutoCustomization does not create a user profile. It is created automatically by the TSO BBI-TS CLIST. A VTAM or EXCP BBI-TS userid.BBPROF must be created manually. If you are using MAINVIEW Alternate Access for EXCP or VTAM communication, you do not need to create a userid.BBPROF library. SBBPROF: Yes, AutoCustomization creates a site profile.	BBI-TS	BBIPROF
<b>Concatenation:</b> User and site-customized libraries should be concatenated before the distributed libraries.				

---

---

# Appendix J OSPI Terminal Definitions for CICS and IMS

The following sections provide information about defining OSPI virtual terminals to CICS and IMS.

## IMS Terminal Definitions

IMS requires all terminals to be defined in the IMS SYSGEN. If an attempt is made to log on with an OSPI ACB that has not been defined to IMS, the session is not accepted. The following example shows the IMS terminal definitions required for OSPI ACBs:

---

```
TERMINAL  MODEL=2 ,
          FEAT=( ,NOCD,NOPEN) ,
          FPBUF=256,OPTIONS=(TRANRESP) ,
          NAME=OSPI0000          ** OSPI ACB name **

          NAME  OSPI0000          ** Any LTERM name **
                                   ** (subject to IMS security) **
```

---

## CICS Terminal Definitions

The terminal definitions required by CICS depend on which features of CICS are used.

If the CICS autoinstall feature is used, local terminal definitions are not required. However, the appropriate terminal models must be defined prior to logon. See “Autoinstall Terminals” for information about defining OSPI terminals to CICS.

If the CICS autoinstall feature is not used, each terminal must be defined prior to logon. See “Explicit Terminal Definitions” for information about defining OSPI terminals to CICS.

### Autoinstall Terminals

If the CICS autoinstall feature is used, terminals need not be defined but the appropriate models must be defined prior to logon. If a model terminal definition that matches the logmode data associated with the OSPI ACB is not available, CICS issues an error message to the transient data queue.

### Explicit Terminal Definitions

If the CICS autoinstall feature is not used, each terminal must be defined before you logon using RDO. If an attempt is made to logon with an OSPI ACB that has not been defined to CICS, the session is not accepted and error messages are written to the transient data queue.

Figure J-1 on page J-3 provides an example of how to use Resource Definition Online to define an OSPI virtual terminal.

## RDO Macro

Figure J-1 Example of Using Resource Definition Online (Part 1 of 2)

```

OVERTYPE TO MODIFY
CEDA Alter
  TYPeterm      : OSPI
  Group         : TEST
RESOURCE TYPE
  DEvice        ==> LUTYPE2
  TERmodel      ==> 2
  SESSiontype   ==>
  LDcllist      :
  SHippable     ==> No                No | Yes
MAPPING PROPERTIES
  PAGesize      ==> 000 , 000        0-999
  ALTPage       ==> 000 , 000        0-999
  ALTSuffix     ==>
  FMhparm       ==> No                No | Yes
  OBOperid      ==> No                No | Yes
PAGING PROPERTIES
  AUTOPage      ==> No                No | Yes
DEVICE PROPERTIES
  DEFscreen     ==> 024 , 080        0-999
  ALTScreen     ==> 000 , 000        0-999
  APLKybd       ==> No                No | Yes
  APLText       ==> No                No | Yes
  AUDiblealarm  ==> No                No | Yes
  COLor         ==> No                No | Yes
  COPy          : No                No | Yes
  DUAlcasekybd ==> No                No | Yes
  EXTendeddds   ==> No                No | Yes
  HILight       ==> No                No | Yes
  Katakana      ==> No                No | Yes
  LIghtpen      ==> No                No | Yes
  Msrcontrol    ==> No                No | Yes
  OBFormat      ==> No                No | Yes
  PARTitions    ==> No                No | Yes
  PRIntadapter  ==> No                No | Yes
  PROgsymbols   ==> No                No | Yes
  VALidation    ==> No                No | Yes
  FORmfeed      ==> No                No | Yes
  HORizform     ==> No                No | Yes
  VERTicalform  ==> No                No | Yes
  TEXTKybd      ==> No                No | Yes
  TEXTPrint     ==> No                No | Yes
  Query         ==> No                No | Cold | All
  OUTline       ==> No                No | Yes
  SOSi          ==> No                No | Yes
  BACKtrans     ==> No                No | Yes
  CGcsgid       ==> 00000 , 00000   0-65535
SESSION PROPERTIES
  ASCii         ==> No                No | 7 | 8
  SENDsize      ==> 00000           0-30720
  RECEivesize   ==> 00000           0-30720
  BRacket       : Yes                Yes | No
  LOGMode       ==> D6327802

```

Figure J-2 Example of Using Resource Definition Online (Part 2 of 2)

---

```

DIAGNOSTIC DISPLAY
ERRLastline ==> No          No | Yes
ERRIntensify ==> No        No | Yes
ERRColor     ==> NO        NO|Blue|Red|Pink|Green
                |Turquoise|Yellow|NEutral
ERRHighlight ==> No        No|Blink|Reverse|Underline

OPERATIONAL PROPERTIES
AUTOConnect ==> No          No | Yes | All
ATi          ==> Yes        No | Yes
TTi          ==> Yes        Yes | No
CReatesess  ==> No          No | Yes
RELreq      ==> No          No | Yes
DIScreq     ==> Yes        Yes | No
Nepclass    ==> 000        0-255
SIGNoff     ==> Yes        Yes | No | Logoff

MESSAGE RECEIVING PROPERTIES
ROutedmsgs  ==> All        All | None | Specific
LOGOnmsg    ==> Yes        No | Yes

APPLICATION FEATURES
BUildchain  : Yes          No | Yes
USerarealen ==> 000        0-255
Ioarealen   ==> 02000 , 02000 0-32767
UCtran      ==> Yes        No | Yes

RECOVERY
RECOVOption ==> Sysdefault  Sysdefault | Clearconv | Releases
                | Uncondrel | None
RECOVNotify ==> None        None | Message | Transaction

```

---

---

---

# Appendix K    Establishing a VTAM Connection for MAINVIEW AutoOPERATOR to PATROL EM Communication

MAINVIEW AutoOPERATOR establishes a VTAM connection to PATROL EM using a unique ACB. The ACB and destination names must be defined to VTAM and to MAINVIEW AutoOPERATOR. For more information, see “Defining VTAM ACBs” and “Modifying the VTAM Connection with BBPARM Member AAOALT00” on page K-5.

To enable MAINVIEW AutoOPERATOR ALERTs to be sent to PATROL EM through a VTAM connection, you must

1. define a VTAM ACB for MAINVIEW AutoOPERATOR to use to communicate with PATROL EM
2. define a 3270 printer to VTAM
3. reset BBPARM member AAOALT00

## Defining VTAM ACBs

You need to define a VTAM ACB that MAINVIEW AutoOPERATOR will use to communicate with PATROL EM; for example:

```
$AONETC APPL AUTH=(ACQ,NOTSO,VPACE)
```

where \$AONETC is a unique user-specified name representing the APPLID keyword specified in the BBPARM member AAOALT00 (see “Modifying the VTAM Connection with BBPARM Member AAOALT00” on page K-5).

You can add this statement to any member in SYS1.VTAMLST. However, if you have other VTAM definitions for the MAINVIEW AutoOPERATOR subsystem, you might want to include this statement in the BBIAPPLx member (refer to the *MAINVIEW Common Customization Guide* for more information about the BBIAPPLx member).

**Note:** Each BBI-SS PAS must have a unique APPLID specified for each subsystem that is connected to PATROL EM. The APPLID name must be different than the one used for BBI-SS PAS to BBI-SS PAS communication.

## Defining 3270 Printers to VTAM

Each PATROL EM outboard processor appears as a 3270 printer (for example, a 3284, 3287, and so on) to VTAM and MAINVIEW AutoOPERATOR. If the connection is SNA, you must define the processor as a LU3 printer and you must not define it as a SCS printer. An example of a VTAM definition for a local non-SNA printer is

---

```
FB4048 LOCAL CUADDR=048 ,  
          DLOGMOD=S3270 ,  
          FEATUR2=MODEL2 .  
          TERM=3286
```

---

An example of a VTAM definition for a local SNA printer is

---

```
A070A08 LU LOCADDR=8 ,  
          DLOGMOD=DSC2K
```

---

## Resetting BBPARM Member AAOALT00

After a VTAM session is established, MAINVIEW AutoOPERATOR communicates ALERT activity to PATROL EM at BBI-SS PAS start up or when you issue the following BBI control commands:

- **.RESET PARM AAOALT00**
- **.RESET NETCALRT**
- **.START COMP NETCALRT**

# Controlling the MAINVIEW AutoOPERATOR to PATROL EM Interface

Table K-1 is a list of BBI control commands that you can use to control the MAINVIEW AutoOPERATOR to PATROL EM interface:

**Table K-1 BBI Control Commands Available to MAINVIEW AutoOPERATOR to PATROL EM Interface for VTAM Connections (Part 1 of 2)**

BBI Control Command	Parameters	Description
.reset aaoalt00	PARM P	<p>drops all connections to PATROL EM and terminates the VTAM session</p> <p>After processing of AAOALTxx where xx can be any suffix, the VTAM connection and all remote connections are established.</p>
.reset netcalrt	[ALL  <i>destid</i> ] [ON OFF RELOAD]	<p>defines any destination that can be stopped or restarted</p> <p>The format of the command is</p> <p><b>.RESET NETC [ALL] <i>destid</i> [ON OFF RELOAD]</b></p> <p>ALL (Default) indicates all destinations are to be processed as specified by ON, OFF, or RELOAD</p> <p><i>destid</i> indicates a specific destination (VTAM ID) to be processed as specified by ON, OFF, or RELOAD</p> <p>ON (Default) attempts to establish a connection</p> <p>OFF requests a disconnect of a connection</p> <p>RELOAD requests that all ALERTs be sent to the specified destination without dropping or reestablishing the connection</p>
.start comp netcalrt	N/A	requests the starting of the MAINVIEW AutoOPERATOR to PATROL EM interface task

**Table K-1 BBI Control Commands Available to MAINVIEW AutoOPERATOR to PATROL EM Interface for VTAM Connections (Part 2 of 2)**

<b>BBI Control Command</b>	<b>Parameters</b>	<b>Description</b>
.stop comp netcalrt	N/A	requests the stopping of a MAINVIEW AutoOPERATOR to PATROL EM interface task
.display netcalrt	<u>ALL</u>   <i>destid</i>	ALL requests the status of all destinations and their current MAINVIEW AutoOPERATOR to PATROL EM interface tasks. It is the default  <i>destid</i> specifies a VTAM ID to request the status of a specific destination

For a complete description of all the available BBI control commands, refer to the *MAINVIEW Administration Guide* where the complete list of BBI control commands is documented.

## Modifying the VTAM Connection with BBPARM Member AAOALT00

Use BBPARM member AAOALT00 to define the VTAM ACB name for sending ALERTs to PATROL EM destinations, the names of the destination devices, and the level of diagnostics issued. The parameters in this member are described in Table K-2.

**Table K-2 BBPARM Member AAOALT00 (Part 1 of 2)**

<b>Parameter</b>	<b>Description</b>
APPLID=name	defines the value of the APPLID keyword on the ACB used by OPEN in establishing the session with VTAM  This value must be the same name as the one entered in the VTAMLIST entry as the name of the application.
DEST=device	defines a VTAM destination for the ALERTs  This VTAM destination (a PATROL EM workstation) must be defined to the network as a 3287 printer.
STATUS= <u>ON</u>  OFF	(optional) defines whether the destination is to be enabled (ON) during initialization, or left disconnected (OFF)  The default is ON.

Table K-2 BBPARM Member AAOALT00 (Part 2 of 2)

Parameter	Description
SYNCINTV= <u>30</u>  nnn	<p>(optional) defines the synchronization interval between MAINVIEW AutoOPERATOR and the workstation</p> <p>At this defined interval, all current ALERTs are retransmitted to the workstation to compensate for any earlier lost transmissions or transmissions that were not processed. The default is 30 minutes.</p>
MSGLVL= <u>WARN</u>  msgtype	<p>(optional) defines which messages are to be written to the journal; where WARN is the default and msgtype can be</p> <ul style="list-style-type: none"> <li>• NONE no messages to Journal</li> <li>• SEVERE no VTAM connection, internal errors</li> <li>• ERROR lost or dropped connection, error limit reached</li> <li>• WARN no BBPARM member, termination request</li> <li>• INFORMATIONAL normal start and stop messages</li> <li>• DEBUG echo all messages in the Journal</li> </ul>

## 3287 Data Stream Format for MAINVIEW AutoOPERATOR to PATROL EM

MAINVIEW AutoOPERATOR sends a message to PATROL EM for each ALERT that is added or deleted. The message consists of entries delimited by one blank space. Refer to Table K-3 on page K-7.

Certain fields (such as **Origin** and **Queue Name**) might contain embedded blanks because they are user-defined. Any embedded blanks are replaced with \_ (underscore). The only exception is the message text where embedded blank spaces are not replaced in this field.

Certain fields (such as **Origin** and **User**) can be left completely blank. Since the fields are position-sensitive, any all-blank field is replaced with a single \_ (underscore).

Table K-3 has a description of each of the fields in the output message.

**Table K-3 Output Messages Description (Part 1 of 2)**

Field Name	Maximum Length	Description
Anchor field	7	<p>literal to identify a message as an ALERT; its value can be</p> <p><b>ALRT2</b> ALERT-add request</p> <p><b>ALRTCX</b> ALERT-cancel request</p> <p>ALERT text is not included in the data stream for ALRTCX.</p> <p><b>ALRTHD2</b> a heading message requesting that all current ALERTs be cancelled, since all ALERTs currently active in the MAINVIEW AutoOPERATOR application are transmitted</p> <p>The ALRTHD2 value is used in the following cases:</p> <ul style="list-style-type: none"> <li>• when MAINVIEW AutoOPERATOR initializes</li> <li>• when any of the following BBI commands are issued:</li> </ul> <p><b>.start comp netcalrt</b> <b>.reset dest</b> <b>.reset aaoalt00</b></p> <ul style="list-style-type: none"> <li>• After each interval specified in BBPARM member AAOALT00 (the SYNCINTV parameter)</li> </ul> <p><b>Note:</b> These values are valid for MAINVIEW AutoOPERATOR version 4.1 only, and they are not valid for earlier versions of MAINVIEW AutoOPERATOR. For earlier versions, refer to the corresponding <i>MAINVIEW AutoOPERATOR Customization Guide</i> edition for that version.</p>
Origin	8	origin of the ALERT
Subsys ID	8	BBI-SS PAS ID
Target	8	target system associated with the ALERT
User	8	user ID of issuer of the ALERT
Date	5	julian date of the ALERT in the form YYDDD
Time	6	time of the ALERT in HHMMSS
Alarm Indicator	1	Y/N - should alarm be sounded

**Table K-3 Output Messages Description (Part 2 of 2)**

Field Name	Maximum Length	Description
Priority	1	user-assigned priority of the ALERT: 1 - Critical 2 - Major 3 - Minor 4 - Warning 5 - Informational 6 - Clearing
Color code	1	user-assigned color of the message: 1 - Red 2 - Pink 3 - Yellow 4 - Dark Blue 5 - Light Blue 6 - Green 7 - White
Queue name	8	name of the MAINVIEW AutoOPERATOR queue to which this ALERT is assigned
Help panel name	8	for MAINVIEW AutoOPERATOR version 4.1 and later, the data stream includes the Help panel name
Key of the ALERT	64	user-specified key of the ALERT
Message text	255	text of the message  <b>Note:</b> The text string /N is to be interpreted as carriage-return line-feed (CR, LF).

## Diagnostics

You can take the following steps when attempting to diagnose whether the VTAM connection has failed.

- Step 1** Verify that PATROL EM is connected to the correct 3x74 port.
- Step 2** If PATROL EM is connected, verify VTAM connectivity and the LUName by connecting a real 3270 terminal.

Once the 3270 terminal is connected, issue the **VARY VTAM** command to inactivate and reactivate the LUName. At the bottom left-hand corner of the display, you should see the Status line. This Status line confirms that you are working with the correct device.

**Step 3** Try issuing the BBI command `.S COMP NETC`

MAINVIEW AutoOPERATOR should write some characters (such as ALRTHD2) to the terminal, which means MAINVIEW AutoOPERATOR is communicating correctly with the terminal.

**Step 4** When you are sure that you can communicate with a 3270 terminal, you can connect a real 3287 (or equivalent) printer. As MAINVIEW AutoOPERATOR ALERTs are being created, they should be written to the printer. (You can further verify that the printer is correctly defined by using CICS or another VTAM application to send output to the printer.)**Step 5** Finally, if MAINVIEW AutoOPERATOR can successfully send ALERTs to the 3287 printer but PATROL EM still cannot receive them, you must begin problem determination steps on the PATROL EM side to determine why PATROL EM is not receiving the ALERTs.



---

# Index

## Symbols

\$EXECJOB F-1  
\$EXECTST F-1  
\$TESTDAT F-1

## A

### AAOALS00

- BBPARAM member E-1
- modifying ALERT parameters, Dynamic Parameter Manager 4-7 to 4-12
- parameters E-6

### AAOALT00

- BBPARAM member E-1
- parameters E-8

### AAOARP00

- BBPARAM member E-2
- modifying Automation Reporter parameters, Dynamic Parameter Manager 4-13 to 4-17
- parameters E-10

### AAOEWD00

- BBPARAM member E-2
- parameters E-12

### AAOEXP00

- description 19-2 to 19-3, E-13
- modifying EXEC parameters
  - Dynamic Parameter Manager 4-17 to 4-29
  - EXEC= 19-2
  - MAXHIGH= 19-2
  - MAXNORM= 19-2

### AAOGME00

- BBPARAM member E-2
- parameters
  - DEBUGMSG E-17
  - GMEACT E-17
  - HOSTNAME E-17
  - LSTNPORT E-17
  - TGTHB E-17
  - TGTIP E-17
  - TGTNAME E-18
  - TGTPORT E-18
  - TGTRTC E-18
  - TGTRTI E-18
  - TRACEAPP E-18
  - TRACEBUF E-18
  - TRACEGME E-19
  - TRACELK E-19

### AAOGME00 parameters

- TGTMXMSL E-17
- TRACESEC E-19
- TRACEZN E-19

### AAOMQL00, BBPARAM member E-2

### AAONCD00

- BBPARAM member E-2
- parameters E-23

### AAOPRM00

- BBPARAM member E-3
- description E-24

### AAORUL00

- BBPARAM member E-3
- parameters E-27

---

AAOTRN00  
 BBPARM member E-3  
 parameters E-29  
 TapeSHARE E-30

AAOTSP00  
 BBPARM member E-3  
 modifying TapeSHARE parameters,  
 Dynamic Parameter Manager 4-30 to 4-38

ACF2 security 11-2

ACTION, modifying AAOTSP00 parameters  
 4-35

address space application, modifying JES2 job  
 numbers 19-4

ALERT QUEUE, AAOARP00 parameter,  
 modifying 4-17

ALERTNV, AAOALS00 parameter, modifying  
 4-11

ALERTs  
 queues  
 creating 16-10  
 deleting 16-10  
 setting storage thresholds, Dynamic  
 Parameter Manager 4-9  
 synchronization 16-9  
 thresholds, Dynamic Parameter Manager 4-7  
 to 4-12

alternate database, CSM application, selecting  
 18-2

AO exit  
 execution 13-9  
 return code 13-9  
 routines 13-7

AOALRTDF F-1

AOALRTPF F-1

AOAPPLDF F-1

AOAPPLPE F-1, F-2

AOCMDDF F-1

AOEEXIT, control statements 13-8

AOEXECDF F-2

AOIEXIT, parameter 13-8

AOPARMDF F-2

ARM  
*See Automatic Restart Manager*

ARMPOLCY parameter, valid values 7-3

AUDITMSG, AAOEXP00 parameter,  
 modifying 4-27

authorization 21-1 to 22-1

AutoCustomization  
 migration between products  
 version 6.1 to 6.3.01 3-15  
 version 6.2 to 6.3.01 2-11  
 version 6.3 to 6.3.01 1-9  
 parameter library I-4  
 procedure library I-8  
 profile library I-7  
 sample library I-5

Automatic Restart Manager (ARM)  
 default policy 7-2  
 implementing support 7-2 to 7-4  
 parameter values 7-3

Automation Reporter parameters, Dynamic  
 Parameter Manager 4-15 to 4-17

AutoOPERATOR  
 BMC Software products 1-6, 2-8, 3-12  
 IBM software requirements 1-5, 2-8, 3-12  
 PATROL EM connection 16-2  
 UBBPROC created for I-8

AutoOPERATOR Access NV  
 checklist 14-2  
 creating  
 DSIPARM member DSICMD 14-4  
 DSIPARM member DSIDMN 14-5  
 DSIPARM member DSIOPF 14-6  
 DSIPARM member NAILOPT00 14-4  
 operator profiles 14-6  
 security 14-7  
 implementing 14-1 to 14-9  
 modifying  
 NAISTUB 14-3  
 NetView JCL 14-3  
 NETVIEW parameter 14-3

AutoRULE, implementing 15-1 to 15-10

## B

BBI Control Command  
 DISPLAY 16-7  
 RESET 16-8  
 START 16-8  
 STOP 16-8

---

BBI-SS PAS to CICS communication  
   available functions 11-7  
   modifying CICS tables 11-3  
   overview 11-1  
   processing program table (PPT) 11-3  
   program control table (PCT) 11-3  
   program list table (PLT) 11-3  
   restarting CICS target regions 11-6  
   Rule Processor 11-7  
   specifying Resources to CICS 11-3  
   startup JCL 11-5  
 BBI-SS PAS, starting before JES 19-5  
 BBI-SS to CICS communication  
   available functions 11-7  
   BROADCAST application 11-7  
 BBIVAR00, BBPARM member E-38  
 BBKEYS, migrating  
   version 6.1 to 6.3.01 3-14  
   version 6.2 to 6.3.01 2-10  
   version 6.3 to 6.3.01 1-7  
 BBLINK  
   deleting old modules 13-2  
   modifying IMS control region JCL 13-6  
 BBPARM  
   AAOALS00 E-1  
   AAOALT00 E-1  
   AAOARP00 E-2  
   AAOEW00 E-2  
   AAOEXP00 E-2  
   AAOGME00 E-2  
   AAOMQL00 E-2  
   AAONCD00 E-2  
   AAOPRM00 E-3  
   AAORUL00 E-3  
   AAOTRN00 E-3  
   AAOTSP00 E-3  
   members E-1 to E-41  
 BBPARM/UBBPARM considerations I-1  
 BBPROF/SBBPROF considerations I-1  
 BBSAMP, members  
   IATUX18 17-4  
   IATUX31 17-4  
   JES2MAO 19-4  
   NAICMDS 14-4  
   NAIINIT 14-6  
   NAIINITX 14-6  
   NAIOPT00 14-4  
   NAIPROF 14-6

BBSAMP, members (*continued*)  
   NAIPROFX 14-6  
   NAISVAR 14-6  
   SSJCL 19-5  
 BBSAMP/UBBSAMP considerations I-1  
 BMC Software products, AutoOPERATOR 1-6,  
   2-8, 3-12  
 BMC Software, contacting ii

## C

CAOTTAB  
   BBSAMP member F-2  
   macro 12-3  
     invoking 12-2  
     parameters 12-4  
 Changed field, Dynamic Parameter Manager 4-6  
 changing transaction IDs, required ZAPs C-2  
 CICSTART F-2  
 CICSTERM F-2  
 Cmd field, Dynamic Parameter Manager 4-6  
 CMRASM 23-1, F-3  
 CMRCSDES F-3  
 CMRDETL (detail performance file) C-2  
 CMRPLT F-3  
 CMRSECU  
   *See also* security  
   action parameters 23-3  
   BBSAMP member F-3  
   implementing 23-1  
   setting 23-3  
   TYPE=ENTRY 23-2  
   TYPE=FINAL 23-2  
   TYPE=INITIAL 23-2  
 CMRSTAT (statistical file) C-2  
 CMRUSECU, BBSAMP member 23-1, F-3  
 command processing 13-7  
 common storage, migrating  
   version 6.1 to 6.3.01 3-11  
   version 6.2 to 6.3.01 2-7  
   version 6.3 to 6.3.01 1-5  
 compatibility, migrating  
   version 6.1 to 6.3.01 3-11  
   version 6.2 to 6.3.01 2-7  
   version 6.3 to 6.3.01 1-5  
 compiled REXX programs 9-1

- configuration
  - data set I-1
  - product libraries I-1
- CONN, GME parameter 16-7
- considerations for implementing CICS 11-2
- consoles
  - allocating 5-1, 5-2
  - CSM application 5-2
  - overview 5-1
  - X-MCS, allocating 5-2, 5-3
- Continuous State Manager
  - accessing for the first time 18-3
  - alternate database, selecting 18-2
  - calculating repository size 18-8
  - creating a shared repository 18-7
  - documentation 1-3
  - End-of-Memory EXEC 18-3
  - EXECs 18-1
  - migrating
    - version 6.1 to 6.3.01 3-3
    - version 6.2 to 6.3.01 2-3
  - performance considerations 18-1
  - Rule Set member, selecting 18-2
- CPO, migrating
  - version 6.1 to 6.3.01 3-14
  - version 6.2 to 6.3.01 2-10
  - version 6.3 to 6.3.01 1-8
- CPU time limit, AutoOPERATOR EXECs, TIMEXLIM 19-2
- CPU usage, migrating
  - version 6.1 to 6.3.01 3-10
  - version 6.2 to 6.3.01 2-7
  - version 6.3 to 6.3.01 1-4
- Created field, Dynamic Parameter Manager 4-6
- CSMALTDDB, SSJCL parameter, setting 18-2
- CSMBUILD 18-1
- CSMDOWN 18-1
- CSMEOM EXEC 18-3
- CSMINIT 18-1
- CSMUP 18-1
- Current field, Dynamic Parameter Manager 4-21
- Current Parm field, Dynamic Parameter Manager 4-39
- Current SSID field, Dynamic Parameter Manager 4-39
- Current Target field, Dynamic Parameter Manager 4-39
- customer support iii

## D

- data set, configuration I-1
- DATAID parameter 11-2
- DB2START F-2
- DB2TERM F-2
- DDNAMEs, migrating
  - version 6.1 to 6.3.01 3-7
  - version 6.2 to 6.3.01 2-5
  - version 6.3 to 6.3.01 1-4
- DEBUGMSG, AAOGME00 parameter, modifying E-17
- Defining AutoOPERATOR Resources to CICS 1-7, 2-10, 3-14
- deleting BBLINK modules 13-2
- dependent services 11-7
- DFHTST TYPE=RECOVERY 11-2
- DFSAOE00 13-7
- DISP, AAOARP00 parameter, modifying 4-16
- DISPLAY command 16-7
- DIVDEF F-2
- DIVUTIL F-2
- DSICMD, DSIPARM member
  - NAICMDS 14-4
  - NAIEXEC 14-4
  - NAISTUB 14-4
- DSIDMN, DSIPARM member, NATASK 14-5
- DSIOPF, DSIPARM member
  - NAIPROF 14-6
  - NAIPROFX 14-6
- DTABJCL F-2
- DUMP data set, BBI-SS PAS 19-5
- Dynamic Parameter Manager
  - activating modifications 4-38 to 4-39
  - fields
    - ALERTNV 4-11
    - AUDITMSG 4-27
    - Changed 4-6
    - Created 4-6
    - Current 4-21
    - EMABCNT 4-21
    - EXABCNT 4-22
    - EXEC 4-28
    - ID 4-6
    - Lib 4-6
    - MAXHIGH 4-22
    - MAXHIGHQ 4-22

---

Dynamic Parameter Manager, fields (*continued*)

- MAXNORM 4-22
- MAXNORMQ 4-22
- MAXSTOR 4-10
- MAXTPUT 4-24
- Member 4-6, 4-21
- OSPINUM 4-27
- OSPIPRFX 4-27
- PEREXLIM 4-26
- PREFIX 4-27
- SELLIM 4-24
- Size 4-6
- SUBXAUTH 4-27
- TIMEXLIM 4-26
- TSOTIME 4-26
- UCPARMS 4-26
- UNITNAME 4-27
- VV.MM 4-6
- WARNLVL1 4-10, 4-23
- WARNLVL2 4-10, 4-23
- modifying ALERTs thresholds in
  - AAOALSxx 4-7 to 4-12
- modifying Automation Reporter parameters in AAOARPxx 4-13 to 4-17
- modifying EXEC parameters in AAOEXPxx 4-17 to 4-29
- modifying TapeSHARE parameters in AAOTSPxx 4-30 to 4-38
- sample panels
  - ALERTs Storage Thresholds 4-9
  - Automation Reporter parameters 4-15
  - AutoOPERATOR Primary Option Menu 4-2
  - Confirm Parameter Activation 4-38
  - EXEC Parms 4-25
  - EXEC Thresholds 4-19
  - setting high-priority EXECs 4-28
  - TapeSHARE parameters 4-32
- setting
  - ALERTs storage thresholds 4-9
  - Automation Reporter parameters 4-15
  - EXEC parameters 4-25
  - EXEC thresholds 4-19
  - high-priority EXECs 4-28
  - TapeSHARE parameters 4-32

## E

- EMABCNT, AAOEXP00 parameter, modifying 4-21
- End-of-Memory (EOM) EXEC, Continuous State Manager 18-3
- EOM (End-of-Memory) EXEC, Continuous State Manager 18-3
- Event Activity Statistics panel, modifying 19-3
- EXABCNT, AAOEXP00 parameter, modifying 4-22
- EXEC
  - AAOEXP00 parameter
    - modifying 4-28, 19-3
    - priority EXECs 19-2
    - threading EXECs 19-2
  - Continuous State Manager 18-1
  - CSMEOM 18-3
  - description I-8
  - migrating
    - version 6.1 to 6.3.01 3-3
    - version 6.2 to 6.3.01 2-2
    - version 6.3 to 6.3.01 1-2
  - parameters, Dynamic Parameter Manager 4-25 to 4-28
  - priority, Dynamic Parameter Manager 4-28
  - REXX, enabling 19-1
  - runaway, loop detection 19-2
  - samples I-3
  - thresholds, Dynamic Parameter Manager 4-19 to 4-24
- exit routines 13-7
- expanding PROFILE pool data set G-1
- Extended MCS consoles
  - See* X-MCS
- EXTRA, CAODTAB macro, parameter 12-4
- extrapartition transient data queue destinations, selected 12-3

---

## F

FCD2 transaction C-2  
FIC2 transaction C-2  
fields  
    ALERTs Storage Thresholds panel 4-12  
    Automation Reporter ParmS panel 4-15 to 4-17  
    Confirm EXEC Update panel 4-39  
    Dynamic Parameter Manager 4-6 to 4-39  
    EXEC PARMS panel 4-21 to 4-28  
    TapeSHARE parameters panel 4-32 to 4-38  
FREE, AAOTSP00 parameter, modifying 4-38  
FST2 QOFF transaction C-2  
FST2 QON transaction C-2

## G

GME keyword  
    displaying connections 16-7  
    displaying local node status 16-7  
    displaying published subjects 16-7  
    displaying trace records 16-7  
    RECEIVE 16-7  
GMEACT, AAOGME00 parameter, modifying E-17

## H

high-priority EXECs, Dynamic Parameter Manager 4-28  
HONORMPF parameter E-24  
HOSTNAME, AAOGME00 parameter, modifying E-17  
HST2 transaction (history display) C-2

## I

IBAOUE3 F-4  
IBM software, AutoOPERATOR requirements 1-5, 2-8, 3-12  
ICOPY7 F-4  
ICOPY8 F-4  
ICOPY9 F-4  
ID field, Dynamic Parameter Manager 4-6

IDEL31\$5 13-2  
IDEL32\$5 13-2  
IDEL32\$6 13-2  
IDEL33\$5 13-2  
IDEL33\$6 13-2  
IDEL33\$7 13-2  
identifying transactions C-2  
IELOAD 13-9  
IMFEXEC CICS commands 11-9  
IMFSYS00 13-8  
IMSID 21-1  
IMSSTART F-2  
IMSTERM F-2  
independent IMFEXEC CICS commands 11-9  
initialization exit 13-7  
INTERVAL, AAOARP00 parameter, modifying 4-16  
INTRA, CAODTAB macro, parameter 12-4  
intrapartition transient data queue destinations, selecting 12-3

## J

JCL samples F-4  
JES, start after AutoOPERATOR 19-5  
JES2 job numbers, displaying 19-4  
JES2MAO, BBSAMP data set 19-4, F-2  
JES3 support 17-1  
JNL2 transaction (subsystem status) C-2

## L

LAST SEG LOST 13-10  
Last Update field, Dynamic Parameter Manager 4-39  
Lib field, Dynamic Parameter Manager 4-6  
libraries  
    configuration I-1  
    distributed target I-1  
    parameter I-3  
    procedure I-3  
    product-user I-1  
    profile I-3  
    sample I-3  
    site-customized I-1

---

LINKNUC F-4  
loop detection, AutoOPERATOR EXECs 19-2  
LSTNPORT, AAOGME00 parameter,  
modifying E-17

## M

MAINVIEW AutoOPERATOR SAP High  
Availability product 1-2  
MAINVIEW Total Object Manager (TOM)  
application 1-2, 1-3  
manual customization  
parameter library I-4  
procedure library I-9  
sample library I-6  
manual start C-2  
manual stop C-2  
MAXHIGH, AAOEXP00 parameter, modifying  
4-22, 19-3  
MAXHIGHQ, AAOEXP00 parameter,  
modifying 4-22  
MAXNORM, AAOEXP00 parameter,  
modifying 4-22, 19-3  
MAXNORMQ, AAOEXP00 parameter,  
modifying 4-22  
MAXSTOR, AAOALS00 parameter, modifying  
4-10  
MAXTPUT, AAOEXP00 parameter, modifying  
4-24  
MCTSIZE=xxxx, specifying 19-3  
Member field, Dynamic Parameter Manager 4-6,  
4-21  
message counter table, defining 19-3  
messages, migrating, version 6.1. to 6.3.01 3-2  
migrating  
version 6.1 to 6.3.01  
AutoCustomization considerations 3-15  
BBKEYS 3-14  
common storage 3-11  
CPO 3-14  
CPU usage 3-10  
DDNAMEs 3-7  
downward compatibility 3-11  
EXECs 3-3  
messages 3-2, 3-3  
parameter values 3-8  
parameters 3-3

migrating, version 6.1 to 6.3.01 (*continued*)  
private storage usage 3-11  
Rules 3-3, 3-5  
security 3-10  
System Maintenance Program (SMP)  
3-16  
upward compatibility 3-11  
variables 3-7  
version 6.2 to 6.3.01  
AutoCustomization considerations 2-11  
BBKEYS 2-10  
common storage 2-7  
CPO 2-10  
CPU usage 2-7  
DDNAMEs 2-5  
downward compatibility 2-7  
EXECs 2-2  
messages 2-3  
parameter values 2-5  
parameters 2-2  
private storage usage 2-7  
Rules 2-2, 2-4  
security 2-6  
System Maintenance Program (SMP)  
2-12  
upward compatibility 2-7  
variables 2-5  
version 6.3 to 6.3.01  
AutoCustomization considerations 1-9  
BBKEYS 1-7  
common storage 1-5  
CPO 1-8  
CPU usage 1-4  
DDNAMEs 1-4  
downward compatibility 1-5  
EXECs 1-2  
parameter values 1-4  
parameters 1-2  
private storage usage 1-5  
Rules 1-2, 1-4  
security 1-4  
System Maintenance Program (SMP)  
1-9  
upward compatibility 1-5  
modifying ALERTs thresholds, Dynamic  
Parameter Manager 4-7 to 4-12

---

modifying Automation Reporter parameters,  
Dynamic Parameter Manager 4-13 to 4-17

modifying EXEC parameters, Dynamic  
Parameter Manager 4-17 to 4-29

modifying TapeSHARE parameters, Dynamic  
Parameter Manager 4-30 to 4-38

MPFFLTR parameter  
description E-24  
implementing 8-2 to 8-5

MQEV, description E-25

MQGINHIB, description E-26

MQNSHARE, description E-26

MQS Rules  
compatibility between 6.1 and 6.3.01 3-5  
compatibility between 6.2 and 6.3.01 2-4  
compatibility between 6.3 and 6.3.01 1-4

MSGLVLI keyword  
BOTH 13-4  
MTO (IMS master terminal) 13-4  
NONE 13-4  
WTO (OS/ 390 system console) 13-4

MTO message capture 13-7

MVS consoles  
*See* consoles, subsystem, X-MCS

MVS subsystem consoles  
*See* consoles, subsystem, X-MCS

## N

NAICMDS F-2

NAIINIT F-2

NAIOPT00  
description F-2  
DSIPARM member, OPERxx statements  
14-4

NAIPROF F-2

NAISTUB, AutoOPERATOR Access NV,  
executing 14-6

NAISVAR F-2

NAITASK F-2

NetView  
DSICLD data set  
NAIINIT 14-6  
NAIINITX 14-6  
NAISVAR 14-6  
JCL, modifying 14-3

NETVIEW parameter, AutoOPERATOR  
Access NV, BBISSP00 14-9

NEW DSN, AAOARP00 parameter, modifying  
4-16

NOGIVE, AAOTSP00 parameter, modifying  
4-35

NOTAKE, AAOTSP00 parameter, modifying  
4-36

## O

OFF INT, AAOARP00 parameter, modifying  
4-16

Open System Procedural Interface (OSPI)  
ACB pool 10-2, 10-3  
accessing VTAM applications 10-1  
defining virtual terminals 10-1  
IMFEXEC LOGON 10-2  
naming conventions 10-2  
OSPINUM 10-3  
OSPIPRFX 10-2  
terminal definitions J-1  
terminal macros J-2

OPID= 23-7

origin 21-1

OSPI  
*See* Open System Procedural Interface  
(OSPI)

OSPINUM, AAOEXP00 parameter, modifying  
4-27

OSPIPRFX, AAOEXP00 parameter, modifying  
4-27

## P

parameter library  
AutoCustomization I-4  
description I-3  
IMS-specific I-4  
manual customization I-4  
use I-3

---

parameters  
  BBPARM E-1  
  migrating  
    version 6.1 to 6.3.01 3-3, 3-8  
    version 6.2 to 6.3.01 2-2, 2-5  
    version 6.3 to 6.3.01 1-2, 1-4  
PARTNER, AAOTSP00 parameter, modifying 4-33  
PATROL EM  
  3270 printers K-2  
  ACB K-1  
  ALERTs 16-9 to 16-10  
  AutoOPERATOR connection 16-2  
  data stream format K-6  
  establishing a VTAM connection K-1  
  implementing TCP/IP interface 16-1  
  implementing VTAM connection 16-1  
  interface  
    controlling 16-7  
    implementing 16-1 to 16-10  
PEREXLIM, AAOEXP00 parameter, modifying 4-26, 19-2  
performance considerations, CSM 18-1  
PM0229W warning message, BBI-SS PAS  
  startup 19-4  
PREF, AAOTSP00 parameter, modifying 4-36  
PREFDEV, AAOTSP00 parameter, modifying 4-37  
PREFIX, AAOEXP00 parameter, modifying 4-27  
private storage usage, migrating  
  version 6.1 to 6.3.01 3-11  
  version 6.2 to 6.3.01 2-7  
  version 6.3 to 6.3.01 1-5  
procedure library  
  AutoCustomization I-8  
  description I-3  
  manual customization I-9  
  use I-8  
processing program table (PPT)  
  modifying 11-3  
  use of C-1  
product libraries  
  configuration I-1  
  site-customized I-1  
  types I-1  
product support iii

profile library  
  AutoCustomization I-7  
  description I-3  
  use I-6  
  user I-1  
PROFILE pool, expanding data set G-1  
program control table (PCT), use of 11-3, C-1  
program list table (PLT) 11-3  
PUBLISH, GME parameter 16-7

## R

RACF security 11-2  
recoverable temporary storage B-1  
Replaced by field, Dynamic Parameter Manager 4-39  
RESET command 16-8  
reset, CMRSECU table 23-1  
RESLIB F-4  
RETRYCNT, AAOTSP00 parameter, modifying 4-34  
RETRYINT, AAOTSP00 parameter, modifying 4-34  
REXX, support for EXECs 19-1  
REXX/370 Alternate Library  
  BBREXALT 9-2  
  implementing 9-1  
  SMP/E messages 9-2  
Rule Set, CSM application, selecting 18-2  
rules processor authorization 22-1  
Rules, migrating  
  version 6.1 to 6.3.01 3-3, 3-5  
  version 6.2 to 6.3.01 2-2, 2-4  
  version 6.3 to 6.3.01 1-2, 1-4  
RULESET parameter E-24  
RVARTST F-2

## S

sample JCL F-4  
sample library  
  AutoCustomization I-5  
  description I-3  
  manual customization I-6  
  use I-5

---

sample members

- \$EXECJOB F-1
- \$EXECTST F-1
- \$TESTDAT F-1
- AOALRTDF F-1
- AOALRTPF F-1
- AOAPPLDF F-1
- AOAPPLPE F-1, F-2
- AOCMDDF F-1
- AOEXECDF F-2
- AOPARMDF F-2
- CAOTTAB F-2
- CICSTART F-2
- CICSTERM F-2
- CMRSECU F-3
- CMRUSECU F-3
- DB2START F-2
- DB2TERM F-2
- DIVDEF F-2
- DIVUTIL F-2
- DTABJCL F-2
- IMSSTART F-2
- IMSTERM F-2
- JES2MAO F-2
- NAICMDS F-2
- NAIINIT F-2
- NAIOPT00 F-2
- NAIPROF F-2
- NAISVAR F-2
- NAITASK F-2
- RVARTST F-2
- TTABJCL F-2

SAP High Availability product 1-2

SBBPROF/BBPROF considerations I-1

security

- CMRASM member 23-1
- CMRSECU macro 23-1
- CMRSECU module 23-1
- CMRUSECU member 23-1
- considerations 11-2
- display parameters 23-3
- implementing 23-1
- migrating
  - version 6.1 to 6.3.01 3-10
  - version 6.2 to 6.3.01 2-6
  - version 6.3 to 6.3.01 1-4
- setting 23-3

security exit, SYSPROG 20-2

security for BBI, BBPARM member 21-1 to 22-1

SELLIM, AAOEXP00 parameter, modifying 4-24

setting ALERTs storage thresholds, Dynamic Parameter Manager 4-9 to 4-12

setting Automation Reporter thresholds, Dynamic Parameter Manager 4-13 to 4-17

setting EXEC parameters, Dynamic Parameter Manager 4-25 to 4-28

setting EXEC thresholds, Dynamic Parameter Manager 4-19 to 4-24

setting high-priority EXECs, Dynamic Parameter Manager 4-28

setting TapeSHARE thresholds, Dynamic Parameter Manager 4-30 to 4-38

Size field, Dynamic Parameter Manager 4-6

SMN2 transaction, manual start C-2

SSJCL, CSMALTDDB parameter, setting 18-2

START command 16-8

STOP command, AutoOPERATOR EXECs 16-8

stopping, AutoOPERATOR EXECs 16-8

storage thresholds, ALERTs, Dynamic Parameter Manager 4-9 to 4-12

SUB=MSTR, SSJCL command 19-6

subsystem consoles

- allocating 5-1, 5-2

- CSM application 5-2

subsystem status C-2

SUBXAUTH, AAOEXP00 parameter, modifying 4-27

support, customer iii

SUPREPLY parameter E-25

syntax notation xxiv

SYS1.HASPSRC, JES2 offsets 19-4

SYSPROG, security exit 20-2

System Maintenance Program (SMP), migrating

- version 6.1 to 6.3.01 3-16

- version 6.2 to 6.3.01 2-12

- version 6.3 to 6.3.01 1-9

---

## T

- TapeSHARE parameters
  - Dynamic Parameter Manager 4-30 to 4-38
    - implementing 6-1
      - activating 6-2
      - password keys 6-2
      - prerequisite 6-1
      - securing 6-2
- TapeSHARE, AAOTSP00, BBPARM member E-30
- TARGET 21-1
- target libraries I-1
- TCP/IP address
  - defining 16-5
  - multiple 16-6
- TCP/IP connection
  - establishing 16-4
  - prerequisites 16-4
- technical support iii
- temporary storage prefix
  - CMRI prefix ZAP B-1
  - existing definition B-1
- Temporary Storage Table (TST)
  - compatibility 11-2
  - use of B-1
- terminal emulation, VTAM sessions, OSPI application 10-1
- terminating, AutoOPERATOR EXECs 16-8
- TGTHB, AAOGME00 parameter, modifying E-17
- TGTIP, AAOGME00 parameter, modifying E-17
- TGTMXMSL, AAOGME00 parameter, modifying E-17
- TGTNAME, AAOGME00 parameter, modifying E-18
- TGTPORT, AAOGME00 parameter, modifying E-18
- TGTRTC, AAOGME00 parameter, modifying E-18
- TGTRTI, AAOGME00 parameter, modifying E-18
- TIMEOUT, AAOTSP00 parameter, modifying 4-33
- TIMEXLIM, AAOEXP00 parameter, modifying 4-26, 19-2
- Total Object Manager (TOM) application 1-3
- TRACE
  - AAOTSP00 parameter, modifying 4-38
  - GME parameter 16-7
- TRACEAPP, AAOGME00 parameter, modifying E-18
- TRACEBUF, AAOGME00 parameter, modifying E-18
- TRACEGME, AAOGME00 parameter, modifying E-19
- TRACELK, AAOGME00 parameter, modifying E-19
- TRACESEC, AAOGME00 parameter, modifying E-19
- TRACEZN, AAOGME00 parameter, modifying E-19
- transaction IDs
  - description C-1
  - FCD2 C-2
  - FIC2 C-2
  - FST2 QOFF C-2
  - FST2 QONN C-2
  - HST2 C-2
  - JNL2 C-2
  - required ZAPs for changed IDs C-2
  - SMN2 C-2
- TRANSEC parameter 11-2
- TSOTIME, AAOEXP00 parameter, modifying 4-26
- TST B-1
- TTABJCL F-2
- type-1 and type-2 AO exit routines 13-7

## U

- UBBPARM/BBPARM considerations I-1
- UBBSAMP/BBSAMP considerations I-1
- UCPARMS, AAOEXP00 parameter, modifying 4-26
- unique message total, modifying 19-3
- UNITNAME, AAOEXP00 parameter, modifying 4-27

---

user

- libraries I-1

- profile I-1

user AO exit

- DFSAOUE0 13-7

- DFSAOUE1 13-7

User field, Dynamic Parameter Manager 4-39

USERID 21-1

## V

variables, migrating

- version 6.1 to 6.3.01 3-7

- version 6.2 to 6.3.01 2-5

VTAM sessions, accessing, OSPI application

- 10-1

VV.MM field, Dynamic Parameter Manager 4-6

## W

WARNLVL1

- AAOALS00 parameter, modifying 4-10

- AAOEXP00 parameter, modifying 4-23

WARNLVL2

- AAOALS00 parameter, modifying 4-10

- AAOEXP00 parameter, modifying 4-23

## X

X-MCS consoles, allocating 5-2, 5-3

XTLOAD utility G-1

## Z

ZAP, CMRI temporary storage prefix B-1

ZAPs for changed transaction IDs C-2

# END USER LICENSE AGREEMENT NOTICE

**BY OPENING THE PACKAGE, INSTALLING, PRESSING "AGREE" OR "YES" OR USING THE PRODUCT, THE ENTITY OR INDIVIDUAL ENTERING INTO THIS AGREEMENT AGREES TO BE BOUND BY THE FOLLOWING TERMS. IF YOU DO NOT AGREE WITH ANY OF THESE TERMS, DO NOT INSTALL OR USE THE PRODUCT, PROMPTLY RETURN THE PRODUCT TO BMC OR YOUR BMC RESELLER, AND IF YOU ACQUIRED THE LICENSE WITHIN 30 DAYS OF THE DATE OF YOUR ORDER CONTACT BMC OR YOUR BMC RESELLER FOR A REFUND OF LICENSE FEES PAID. IF YOU REJECT THIS AGREEMENT, YOU WILL NOT ACQUIRE ANY LICENSE TO USE THE PRODUCT.**

This Agreement ("**Agreement**") is between the entity or individual entering into this Agreement ("You") and BMC Software Distribution, Inc., a Delaware corporation located at 2101 CityWest Blvd., Houston, Texas, 77042, USA or its affiliated local licensing entity ("BMC"). "You" includes you and your Affiliates. "Affiliate" is defined as an entity which controls, is controlled by or shares common control with a party. IF MORE THAN ONE LICENSE AGREEMENT COULD APPLY TO THE PRODUCT, THE FOLLOWING ORDER OF LICENSE AGREEMENT PRECEDENCE APPLIES: (1) WEB BASED LICENSE AGREEMENT WITH BMC, (2) WRITTEN LICENSE AGREEMENT WITH BMC, (3) SHRINK-WRAP LICENSE AGREEMENT WITH BMC PROVIDED WITH THE PRODUCT, AND (4) THIS ELECTRONIC LICENSE AGREEMENT WITH BMC. In addition to the restrictions imposed under this Agreement, any other usage restrictions contained in the Product installation instructions or release notes shall apply to Your use of the Product.

**PRODUCT AND CAPACITY.** "**Software**" means the object code version of the computer programs provided, via delivery or electronic transmission, to You. Software includes computer files, enhancements, maintenance modifications, upgrades, updates, bug fixes, and error corrections.

"**Documentation**" means all written or graphical material provided by BMC in any medium, including any technical specifications, relating to the functionality or operation of the Software.

"**Product**" means the Software and Documentation.

"**License Capacity**" means the licensed capacity for the Software with the pricing and other license defining terms, including capacity restrictions, such as tier limit, total allowed users, gigabyte limit, quantity of Software, and/or other capacity limitations regarding the Software. For licenses based on the power of a computer, You agree to use BMC's current computer classification scheme, which is available at <http://www.bmc.com> or can be provided to You upon request.

**ACCEPTANCE.** The Product is deemed accepted by You, on the date that You received the Product from BMC.

**LICENSE.** Subject to the terms of this Agreement, as well as Your payment of applicable fees, BMC grants You a non-exclusive, non-transferable, perpetual (unless a term license is provided on an order) license for each copy of the Software, up to the License Capacity, to do the following:

- (a) install the Software on Your owned or leased hardware located at a facility owned or controlled by You in the country where You acquired the license;
- (b) operate the Software solely for processing Your own data in Your business operations; and
- (c) make one copy of the Software for backup and archival purposes only (collectively a "**License**").

If the Software is designed by BMC to permit you to modify such Software, then you agree to only use such modifications or new software programs for Your internal purposes or otherwise consistent with the License. BMC grants You a license to use the Documentation solely for Your internal use in Your operations.

**LICENSE UPGRADES.** You may expand the scope of the License Capacity only pursuant to a separate agreement with BMC for such expanded usage and Your payment of applicable fees. There is no additional warranty period or free support period for license upgrades.

**RESTRICTIONS:** You agree to **NOT**:

- (a) disassemble, reverse engineer, decompile or otherwise attempt to derive any Software from executable code;
- (b) distribute or provide the Software to any third party (including without limitation, use in a service bureau, outsourcing environment, or processing the data of third parties, or for rental, lease, or sublicense); or
- (c) provide a third party with the results of any functional evaluation or benchmarking or performance tests, without BMC's prior written approval, unless prohibited by local law.

**TRIAL LICENSE.** If, as part of the ordering process, the Product is provided on a trial basis, then these terms apply: (i) this license consists solely of a non-exclusive, non-transferable evaluation license to operate the Software for the period of time specified from BMC or, if not specified, a 30 day time period ("**Trial Period**") only for evaluating whether You desire to acquire a capacity-based license to the Product for a fee; and (ii) Your use of the Product is on an AS IS basis without any warranty, and **BMC, ITS AFFILIATES AND RESELLERS, AND LICENSORS DISCLAIM ANY AND ALL WARRANTIES (INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT) AND HAVE NO LIABILITY WHATSOEVER RESULTING FROM THE USE OF THIS PRODUCT UNDER THIS TRIAL LICENSE ("Trial License").** BMC may terminate for its convenience a Trial License upon notice to You. When the Trial Period ends, Your right to use this Product automatically expires. If You want to continue Your use of the Product beyond the Trial Period, contact BMC to acquire a capacity-based license to the Product for a fee.

**TERMINATION.** This Agreement shall immediately terminate if You breach any of its terms. Upon termination, for any reason, You must uninstall the Software, and either certify the destruction of the Product or return it to BMC.

**OWNERSHIP OF THE PRODUCT.** BMC or its Affiliates or licensors retain all right, title and interest to and in the BMC Product and all intellectual property, informational, industrial property and proprietary rights therein. BMC neither grants nor otherwise transfers any rights of ownership in the BMC Product to You. Products are protected by applicable copyright, trade secret, and industrial and intellectual property laws. BMC reserves any rights not expressly granted to You herein.

**CONFIDENTIAL AND PROPRIETARY INFORMATION.** The Products are and contain valuable confidential information of BMC (“**Confidential Information**”). Confidential Information means non-public technical and non-technical information relating to the Products and Support, including, without limitation, trade secret and proprietary information, and the structure and organization of the Software. You may not disclose the Confidential Information to third parties. You agree to use all reasonable efforts to prevent the unauthorized use, copying, publication or dissemination of the Product.

**WARRANTY.** Except for a Trial License, BMC warrants that the Software will perform in substantial accordance with the Documentation for a period of one year from the date of the order. This warranty shall not apply to any problems caused by software or hardware not supplied by BMC or to any misuse of the Software.

**EXCLUSIVE REMEDY.** BMC’s entire liability, and Your exclusive remedy, for any defect in the Software during the warranty period or breach of the warranty above shall be limited to the following: BMC shall use reasonable efforts to remedy defects covered by the warranty or replace the defective Software within a reasonable period of time, or if BMC cannot remedy or replace such defective copy of the Software, then BMC shall refund the amount paid by You for the License for that Software. BMC’s obligations in this section are conditioned upon Your providing BMC prompt access to the affected Software and full cooperation in resolving the claim.

**DISCLAIMER. EXCEPT FOR THE EXPRESS WARRANTIES ABOVE, THE PRODUCT IS PROVIDED “AS IS.” BMC, ITS AFFILIATES AND LICENSORS SPECIFICALLY DISCLAIM ALL OTHER WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. BMC DOES NOT WARRANT THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR FREE, OR THAT ALL DEFECTS CAN BE CORRECTED.**

**DISCLAIMER OF DAMAGES. IN NO EVENT IS BMC, ITS AFFILIATES OR LICENSORS LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES RELATING TO OR ARISING OUT OF THIS AGREEMENT, SUPPORT, AND/OR THE PRODUCT (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, LOST COMPUTER USAGE TIME, AND DAMAGE OR LOSS OF USE OF DATA), EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, AND IRRESPECTIVE OF ANY NEGLIGENCE OF BMC OR WHETHER SUCH DAMAGES RESULT FROM A CLAIM ARISING UNDER TORT OR CONTRACT LAW.**

**LIMITS ON LIABILITY. BMC’S AGGREGATE LIABILITY FOR DAMAGES IS LIMITED TO THE AMOUNT PAID BY YOU FOR THE LICENSE TO THE PRODUCT.**

**SUPPORT.** If Your order includes support for the Software, then BMC agrees to provide support (24 hours a day/7 days a week) (“**Support**”). You will be automatically re-enrolled in Support on an annual basis unless BMC receives notice of termination from You as provided below. There is a free support period during the one year warranty period.

(a) **Support Terms.** BMC agrees to make commercially reasonable efforts to provide the following Support: (i) For malfunctions of supported versions of the Software, BMC provides bug fixes, patches or workarounds in order to cause that copy of the Software to operate in substantial conformity with its then-current operating specifications; and (ii) BMC provides new releases or versions, so long as such new releases or versions are furnished by BMC to all other enrolled Support customers without additional charge. BMC may refuse to provide Support for any versions or releases of the Software other than the most recent version or release of such Software made available by BMC. Either party may terminate Your enrollment in Support upon providing notice to the other at least 30 days prior to the next applicable Support anniversary date. If You re-enroll in Support, BMC may charge You a reinstatement fee of 1.5 times what You would have paid if You were enrolled in Support during that time period.

(b) **Fees.** The annual fee for Support is 20% of the Software’s list price less the applicable discount or a flat capacity based annual fee. BMC may change its prices for the Software and/or Support upon at least 30 days notice prior to Your support anniversary date.

**VERIFICATION.** If requested by BMC, You agree to deliver to BMC periodic written reports, whether generated manually or electronically, detailing Your use of the Software in accordance with this Agreement, including, without limitation, the License Capacity. BMC may, at its expense, perform an audit, at your facilities, of Your use of the Software to confirm Your compliance with the Agreement. If an audit reveals that You have underpaid fees, You agree to pay such underpaid fees. If the underpaid fees exceed 5% of the fees paid, then You agree to also pay BMC’s reasonable costs of conducting the audit.

**EXPORT CONTROLS.** You agree not to import, export, re-export, or transfer, directly or indirectly, any part of the Product or any underlying information or technology except in full compliance with all United States, foreign and other applicable laws and regulations.

**GOVERNING LAW.** This Agreement is governed by the substantive laws in force, without regard to conflict of laws principles: (a) in the State of New York, if you acquired the License in the United States, Puerto Rico, or any country in Central or South America; (b) in the Province of Ontario, if you acquired the License in Canada (subsections (a) and (b) collectively referred to as the “**Americas Region**”); (c) in Singapore, if you acquired the License in Japan, South Korea, Peoples Republic of China, Special Administrative Region of Hong Kong, Republic of China, Philippines, Indonesia, Malaysia, Singapore, India, Australia, New Zealand, or Thailand (collectively, “**Asia Pacific Region**”); or (d) in the Netherlands, if you acquired the License in any other country not described above. The United Nations Convention on Contracts for the International Sale of Goods is specifically disclaimed in its entirety.

**ARBITRATION. ANY DISPUTE BETWEEN YOU AND BMC ARISING OUT OF THIS AGREEMENT OR THE BREACH OR ALLEGED BREACH, SHALL BE DETERMINED BY BINDING ARBITRATION CONDUCTED IN ENGLISH. IF THE DISPUTE IS INITIATED IN THE AMERICAS REGION, THE ARBITRATION SHALL BE HELD IN NEW YORK, U.S.A., UNDER THE CURRENT COMMERCIAL OR INTERNATIONAL, AS APPLICABLE, RULES OF THE AMERICAN ARBITRATION ASSOCIATION. IF THE DISPUTE IS INITIATED IN A COUNTRY IN THE ASIA PACIFIC REGION, THE ARBITRATION SHALL BE HELD IN SINGAPORE, SINGAPORE UNDER THE CURRENT UNCITRAL ARBITRATION RULES. IF THE DISPUTE IS INITIATED IN A COUNTRY OUTSIDE OF THE AMERICAS REGION OR ASIA PACIFIC REGION, THE ARBITRATION SHALL BE HELD IN AMSTERDAM, NETHERLANDS UNDER THE CURRENT UNCITRAL ARBITRATION RULES. THE COSTS OF THE ARBITRATION SHALL BE BORNE EQUALLY PENDING THE ARBITRATOR’S AWARD. THE AWARD RENDERED SHALL BE FINAL AND BINDING UPON THE PARTIES AND SHALL NOT BE SUBJECT TO APPEAL TO ANY COURT, AND MAY BE ENFORCED IN ANY COURT OF COMPETENT JURISDICTION. NOTHING IN THIS AGREEMENT SHALL BE DEEMED AS PREVENTING EITHER PARTY FROM SEEKING INJUNCTIVE RELIEF FROM ANY COURT HAVING JURISDICTION OVER THE PARTIES**

**AND THE SUBJECT MATTER OF THE DISPUTE AS NECESSARY TO PROTECT EITHER PARTY'S CONFIDENTIAL INFORMATION, OWNERSHIP, OR ANY OTHER PROPRIETARY RIGHTS. ALL ARBITRATION PROCEEDINGS SHALL BE CONDUCTED IN CONFIDENCE, AND THE PARTY PREVAILING IN ARBITRATION SHALL BE ENTITLED TO RECOVER ITS REASONABLE ATTORNEYS' FEES AND NECESSARY COSTS INCURRED RELATED THERETO FROM THE OTHER PARTY.**

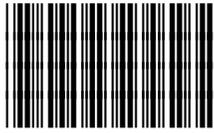
**U.S. GOVERNMENT RESTRICTED RIGHTS.** The Software under this Agreement is "commercial computer software" as that term is described in 48 C.F.R. 252.227-7014(a)(1). If acquired by or on behalf of a civilian agency, the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of this Agreement as specified in 48 C.F.R. 12.212 (Computer Software) and 12.211 (Technical Data) of the Federal Acquisition Regulations ("**FAR**") and its successors. If acquired by or on behalf of any agency within the Department of Defense ("**DOD**"), the U.S. Government acquires this commercial computer software and/or commercial computer software documentation subject to the terms of this Agreement as specified in 48 C.F.R. 227.7202 of the DOD FAR Supplement and its successors.

**MISCELLANEOUS TERMS.** You agree to pay BMC all amounts owed no later than 30 days from the date of the applicable invoice, unless otherwise provided on the order for the License to the Products. You will pay, or reimburse BMC, for taxes of any kind, including sales, use, duty, tariffs, customs, withholding, property, value-added (VAT), and other similar federal, state or local taxes (other than taxes based on BMC's net income) imposed in connection with the Product and/or the Support. This Agreement constitutes the entire agreement between You and BMC and supersedes any prior or contemporaneous negotiations or agreements, whether oral, written or displayed electronically, concerning the Product and related subject matter. No modification or waiver of any provision hereof will be effective unless made in a writing signed by both BMC and You. You may not assign or transfer this Agreement or a License to a third party without BMC's prior written consent. Should any provision of this Agreement be invalid or unenforceable, the remainder of the provisions will remain in effect. The parties have agreed that this Agreement and the documents related thereto be drawn up in the English language. Les parties exigent que la présente convention ainsi que les documents qui s'y rattachent soient rédigés en anglais.

SW Click EULA 071102



# Notes



\*42110\*