

MAINVIEW[®] SRM Reporting Reference Manual

**MAINVIEW[®] SRM StorageGUARD
MAINVIEW[®] SRM SG-Control
MAINVIEW[®] SRM EasyHSM**

Version 7.2

June 20, 2003



Copyright 2003 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. IBM and DB2 are registered trademarks of International Business Machines Corp. All other registered trademarks or trademarks belong to their respective companies.

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

Restricted Rights Legend

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b) or DFARS 227.7202, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

Contacting BMC Software

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

United States and Canada

Address BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827

Telephone 713 918 8800 or
800 841 2031

Fax 713 918 8000

Outside United States and Canada

Telephone (01) 713 918 8800

Fax (01) 713 918 8000

Customer Support

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

Support Web Site

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

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by Telephone or E-mail

In the United States and Canada, if you need technical support and do not have access to the Web, call 800 537 1813. Outside the United States and Canada, please contact your local support center for assistance. To find telephone and e-mail contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at www.bmc.com/support.html.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem immediately:

- product information
 - product name
 - product version (release number)
 - license number and password (trial or permanent)
- operating system and environment information
 - machine type
 - operating system type, version, and service pack or other maintenance level such as PUT or PTF
 - system hardware configuration
 - serial numbers
 - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
 - product error messages
 - messages from the operating system, such as `file system full`
 - messages from related software

Contents

About This Book	xiii
Chapter 1	Introduction to MAINVIEW SRM Reporting
Chapter 2	Reporting Functions
	Overview..... 2-2
	Application Collector Function 2-2
	System Parameters 2-3
	Rule List Parameter Reference..... 2-4
	Parameter Explanations 2-5
	Supported Filter List Parameters 2-6
	Performance Collector Functions 2-9
	HSM Collector Functions..... 2-17
	HSMBACKP—Control DFHSM Backups 2-17
	HSMDELET—Enhance DFHSM Deletion 2-21
	HSMGCCNV—Apply Calendar Conversion to DFHSM Migration 2-23
	HSMIGRT—Enhance DFHSM Migration..... 2-27
	HSMRECAL—Pool Data Sets During DFHSM Recall 2-31
	SMHSMUTL—DFHSM Control Data Set Batch Utility 2-36
Chapter 3	Space Collector
	Overview..... 3-2
	How the Space Collector Works..... 3-2
	Using the Copy/Merge Utility 3-3
	Copy/Merge Processing Options 3-3
	Controlling the Space Collector 3-6
	Activating and Deactivating the Space Collector 3-8
	Using More Than One Space Collector 3-10
	Implementing Security 3-11
Chapter 4	Performance Collector
	Overview..... 4-2
	How the Performance Collector Works 4-2
	Activating SMF and CMF/RMF Parameters 4-2

CMF Requirements for Data Collection	4-3
RMF Requirements for Data Collection	4-4
SMF Requirements for Data Collection	4-4
Controlling the Performance Collector	4-5
Filtering Performance Data	4-6

Chapter 5

Application Collector

How the Application Collector Works	5-2
Check Processing	5-2
Add Processing	5-4
Modes of Operation	5-4
Combining Modes	5-5
Creating Applications	5-5
Dynamically Creating an Application	5-6
Manually Creating an Application	5-7
Maintaining Budgets	5-7
Assigning Applications to a Data Set	5-8
Reviewing and Updating Applications	5-8
Reporting	5-9
TSO BUDGET Command as a Batch TMP	5-10
DFHSM Migration and Backup Report	5-11
SAS Reporting Procedures	5-12
COBOL Reporting Programs	5-13
Reporting with the SGCCOB01 Member	5-13
Trend Analysis Reporting	5-17
TSO Commands	5-18
BUDDSN Command	5-18
BUDGET Command	5-21

Chapter 6

HSM Collector

How the HSM Collector Works	6-2
DFHSM Control Data Sets	6-2
DFHSM Log Extract Process	6-3
Automated Processing	6-4
Batch Processing	6-5
Log Extract Data Set	6-7
User Job Execution	6-7
Sysplex Environments with Shared DASD Considerations	6-8
System Parameters	6-8
DFHSM Output Management	6-14
Message Filtering	6-15
Input Data Sets	6-16
Output Data Sets	6-17
Skeleton Processing	6-28
DFHSM Output Management Views	6-34

Chapter 7

VTOC Scan Processing

How VTOC Scan Processing Works	7-2
--	-----

	Initiating a VTOC Scan	7-3
	SMVSCFxx Filter Member	7-3
	SMMSYSxx member	7-4
	VTOC Scan Extract File	7-8
Chapter 8	Tape Reporting	
	Overview	8-2
	Tape Reporting Features	8-2
	Supported Environments	8-5
	Tape Reports	8-5
	Initiating an SVOS TSCAN	8-5
	System Parameters	8-6
Appendix A	System Exits	
	Overview	A-2
	Installing Exits	A-2
	Exit Flow Example	A-3
	Security Exit	A-4
	Controlling Access to Applications	A-5
	Controlling Application Data	A-5
	Controlling Performance	A-5
	Register Conventions	A-6
	Application Code Build Exit	A-7
	Register Conventions	A-8
	Add Exit	A-9
	Register Conventions	A-9
	Check Exit	A-10
	Register Conventions	A-11
	Default Exit	A-12
	Budget Models	A-12
	Register Conventions	A-13
	Select Exit	A-13
	Register Conventions	A-14
	Table of Sample Exits	A-15
	Application Records	A-16
	Global Work Area	A-18
	Special Considerations for Exits	A-18
	DFHSM Exit Routines	A-19
Appendix B	Application Database Utilities	
	Overview	B-2
	Initializing and Updating the Database	B-2
	Control Statements	B-2
	Comments	B-3
	Resynchronizing the Database	B-12
	SGCMAINT RESYNC Statement	B-13
	SGCRSYNC Multitasking RESYNC	B-16
	Diagnostic Reporting	B-17

Application Database Reports	B-17
Application Database Compare	B-20

Index

List of Figures

Figure 5-1	Check Processing	5-3
Figure 5-2	Add Processing	5-4
Figure 5-3	BUDGET TSO Command Batch TMP	5-11
Figure 5-4	Sample Output	5-11
Figure 5-5	DFHSM Migration and Backup Totals	5-12
Figure 5-6	Database Detail Report	5-14
Figure 5-7	Top Five Users of Permanent and VSAM Space	5-15
Figure 5-8	Applications over Warning Threshold Report	5-15
Figure 5-9	Summary of High-Water Marks	5-16
Figure 5-10	Application Graph	5-16
Figure 5-11	Alphabetic Listing of Applications	5-16
Figure 5-12	Database Trend Report	5-17
Figure 6-1	Example JCL that Extracts Data from Two Log Files	6-6
Figure 6-2	Sample JCL for Output Management (member JCLOPM in the BBSAMP library	6-20
Figure 6-3	DFHSM Output Management Action and Selection Parameters Overview	6-23
Figure 2-1	SGCMAINT JCL	B-2
Figure B-2	Sample Output - Example 1	B-19
Figure B-3	Sample Output - Example 2	B-19

List of Tables

Table 1-1	Subcomponent Cross Reference	1-1
Table 2-1	System Parameters Specific to the SGCONTL Function	2-3
Table 2-2	System Parameters Common to MAINVIEW SRM	2-3
Table 2-3	Rule List Parameter Reference	2-4
Table 2-4	Performance Collector Functions	2-9
Table 2-5	Filter List Parameters for Performance and Space Collectors	2-14
Table 2-6	Parameter Quick Reference	2-18
Table 2-7	Parameter Quick Reference	2-23
Table 2-8	Parameter Quick Reference	2-27
Table 2-9	Parameter Quick Reference	2-32
Table 2-10	Parameter Quick Reference	2-36
Table 3-1	Space Collector System Parameters	3-7
Table 3-2	Requirements for SCDCOLLS DD Statements	3-8
Table 4-1	Required Keywords in CMFCPMxx	4-3
Table 4-2	Performance Collector System Parameters	4-5
Table 5-1	Effects of Modes on Application Processing	5-5
Table 5-2	Field Table for BUDGET Command	5-25
Table 6-1	System Parameters	6-8
Table 6-2	System Parameters	6-18
Table 6-3	DD Statement Descriptions	6-21
Table 6-4	Action Parameters	6-26
Table 6-5	Selection Parameters	6-26
Table 6-6	Output Parameters	6-27
Table 6-7	Message Inclusion Parameters	6-27
Table 6-8	Action Parameters Used in Skeleton Processing	6-29
Table 6-9	DFHSM Output Management Views	6-34
Table 7-1	Filter Keywords in the SMVSCFxx Member	7-3
Table 7-2	Age Range Parameters	7-5
Table 7-3	Size Range Parameters	7-6
Table 7-4	Percentage-Used Parameters	7-7
Table 8-1	Interrogated Software	8-3
Table 8-2	Software Usage Categories	8-4
Table 8-3	System Parameters	8-6
Table A-1	System Exits	A-2

Table A-2	Security Exit Register Contents	A-6
Table A-3	Application Code Build Exit Register Contents	A-8
Table A-4	Add Exit Register Contents	A-9
Table A-5	Check Exit Register Contents	A-11
Table A-6	Default Exit Register Contents	A-13
Table A-7	Select Exit Register Contents	A-14
Table A-8	Sample Exits	A-15
Table A-9	Application Record Format	A-16
Table B-1	Application Code Fields	B-11

About This Book

This book contains reference information about the MAINVIEW® Storage Resource Manager Reporting products by BMC Software and is intended for storage administrators.

To use this book, you should be familiar with the following items:

- OS/390 operating system
- job control language (JCL)
- Interactive System Productivity Facility (ISPF)
- MAINVIEW SRM operations (see the *MAINVIEW SRM User Guide and Reference*)
- MAINVIEW infrastructure and navigation (see *Using MAINVIEW*)

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, an index appears at the end of the book.

Chapter/Appendix	Description
Chapter 1, "Introduction to MAINVIEW SRM Reporting"	provides a cross-reference from product names to subcomponent names and basic product operation
Chapter 2, "Reporting Functions"	provides information about functions used by the HSM, performance, and application collectors
Chapter 3, "Space Collector"	describes how the space collector works
Chapter 4, "Performance Collector"	describes how the performance collector works
Chapter 5, "Application Collector"	describes how the application collector works

Chapter/Appendix	Description
Chapter 6, "HSM Collector"	describes how the HSM collector works
Chapter 7, "VTOC Scan Processing"	describes how the VTOC scan works
Chapter 8, "Tape Reporting"	describes how tape reporting works
Appendix A, "System Exits"	explains how to install exits that allow you to extend the functions of the product
Appendix B, "Application Database Utilities"	describes the SGCMAINT and SGCRSYNC batch utility programs

Related Documentation

BMC Software products are supported by several types of documentation:

- online and printed books
- online Help
- release notes and other notices

In addition to this book and the online Help, you can find useful information in the publications listed in the following table. These publications are available on request from BMC Software.

Category	Document	Description
general	<i>MAINVIEW Products General Information</i>	provides an overview of the MAINVIEW environment and the products that it supports
MAINVIEW common documents	<i>OS/390 and z/OS Installer Guide</i> <i>MAINVIEW Installation Requirements Guide</i> <i>MAINVIEW Common Customization Guide</i> <i>Using MAINVIEW</i> <i>MAINVIEW Administration Guide</i> <i>Implementing Security for MAINVIEW</i>	provide instructions for installing, configuring, using, and administering MAINVIEW
<i>MAINVIEW SRM</i> customization documents	<i>MAINVIEW SRM Customization Guide</i>	provides instructions for configuring and customizing MAINVIEW SRM
core documents	<i>MAINVIEW SRM User Guide and Reference</i>	provides information common to all MAINVIEW SRM products and high-level navigation
	<i>MAINVIEW SRM Reference Summary</i>	provides a reference of global parameters, filter list and rule list parameters, and functions
user document	<i>MAINVIEW SRM Reporting User Guide</i>	provides how-to information about MAINVIEW SRM Reporting

Category	Document	Description
messages	<i>MAINVIEW SRM Messages</i>	provides hardcopy of messages that are also available online
supplemental documents	release notes, flashes, technical bulletins	provides additional information about the product

Online and Printed Books

The books that accompany BMC Software products are available in online format and printed format. If you are a Windows or Unix user, you can view online books with Acrobat Reader from Adobe Systems. The reader is provided at no cost, as explained in “To Access Online Books.” You can also obtain additional printed books from BMC Software, as explained in “To Request Additional Printed Books.”

To Access Online Books

Online books are formatted as Portable Document Format (PDF) files. You can view them, print them, or copy them to your computer by using Acrobat Reader 3.0 or later. You can access online books from the documentation compact disc (CD) that accompanies your product or from the World Wide Web.

For information about downloading the free reader from the Web, go to the Adobe Systems site at <http://www.adobe.com>.

To view any online book that BMC Software offers, visit the support page of the BMC Software Web site at <http://www.bmc.com/support.html>. Select a product to access the related documentation.

To Request Additional Printed Books

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

Release Notes and Other Notices

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

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

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

Conventions

This section provides examples of the conventions used in this book and explains how to read ISPF panel-flow diagrams and syntax statements.

General Conventions

This book uses the following general conventions:

Item	Example
information that you are instructed to type	Type SEARCH DB in the designated field. Type search db in the designated field. (Unix)
specific (standard) keyboard key names	Press Enter .
GUI elements and menu sequences	Choose File => Open .
directories, file names, Web addresses, e-mail addresses	The BMC Software home page is at www.bmc.com .
code examples, syntax statements, system messages, screen text	//STEPLIB DD The table <i>tableName</i> is not available.
emphasized words, new terms	The instructions that you give to the software are called <i>commands</i> .
variables	In this message, the variable <i>fileName</i> represents the file that caused the error.

This book uses the following types of special text:

Note: Notes contain important information that you should consider.

Warning! Warnings alert you to situations that could cause problems, such as loss of data, if you do not follow instructions carefully.

Tip: Tips contain information that might improve product performance or that might make procedures easier to follow.

Syntax Statements

Syntax statements appear in the Courier typeface. The following example shows a sample syntax statement:

```
COMMAND KEYWORD1 [KEYWORD2|KEYWORD3] KEYWORD4={YES|NO}
      file_name...
```

The following table explains conventions for syntax statements and provides examples:

Item	Example
Items in italic type represent variables that you must replace with a name or value.	<code>dtsbackup <i>control_directory</i></code>
Brackets indicate a group of options. You can choose at least one of the items in the group, but none of them is required. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option.	<code>[<i>table_name, column_name, field</i>]</code>
Braces enclose a list of required items. You must enter at least one of the items. Do not type the braces when you enter the item.	<code>{<i>DBD_name table_name</i>}</code>
A vertical bar means that you can choose only one of the listed items. In the example, you would choose either <i>commit</i> or <i>cancel</i> .	<code>{<i>commit cancel</i>}</code>
An ellipsis indicates that you can repeat the previous item or items as many times as necessary.	<code><i>column_name . . .</i></code>

Chapter 1 Introduction to MAINVIEW SRM Reporting

MAINVIEW SRM Reporting is a component product grouping of MAINVIEW SRM, which includes three powerful reporting products:

- MAINVIEW SRM EasyHSM
- MAINVIEW SRM StorageGUARD
- MAINVIEW SRM SG-Control

Table 1-1 lists the subcomponents by product that are available in the Reporting component.

Table 1-1 Subcomponent Cross Reference

Product	Reporting Subcomponents
EasyHSM	HSM collector
StorageGUARD	space collector performance collector VTOC scan tape reporting
SG-Control	application collector

When you buy one product in the grouping, you are authorized to use all three. This book provides reference material to supplement the *MAINVIEW SRM Reporting User Guide*.

For a description of the interface and system navigation instructions, refer to the *MAINVIEW SRM User Guide and Reference*. If you are migrating from a previous release of EasyHSM, StorageGUARD, or SG-Control, see the *MAINVIEW SRM Customization Guide*.

Chapter 2 Reporting Functions

This chapter presents the following topics:

Overview	2-2
Application Collector Function	2-2
Performance Collector Functions	2-9
HSM Collector Functions	2-17
HSMBACKP—Control DFHSM Backups	2-17
HSMDELET—Enhance DFHSM Deletion	2-21
HSMCCNV—Apply Calendar Conversion to DFHSM Migration	2-23
HSMIGRT—Enhance DFHSM Migration	2-27
HSMRECAL—Pool Data Sets During DFHSM Recall	2-31
SMHSMUTL—DFHSM Control Data Set Batch Utility	2-36

Overview

In the MAINVIEW SRM suite of products, storage management services are divided into functions. Functions provide all of the runtime services for the application collector, the performance collector, and the HSM collector.

Functions are defined in the SMFUNC xx parmlib member. SMFUNC xx is read during system start up. The suffix specification in the SMMSYS xx member indicates which version of SMFUNC xx contains function specifications for the particular configuration that is being executed.

SMFUNC xx points to the SMFLST xx (filter list) member, which selects resources and controls the operation of the functions. SMFUNC xx is a required member, and with the exception of the space collector, MAINVIEW SRM does nothing without defined function parameters. For more information about system and function definition, see the *MAINVIEW SRM User Guide and Reference*.

You can change parameters using the Functions option on the Parmlib Members menu, which you access from the EZSRM Menu.

Application Collector Function

The SGCONTRL function controls applications. The SGCONTRL function monitors space at allocation and deallocation using the DADSM exits (preprocessing and postprocessing) and any existing DADSM exits you may have established. Only user-defined space requests are tracked. Individual budgetary applications (logical groups) have preestablished space budgets (thresholds) and are updated dynamically. Space requests can be approved or denied at allocation.

Applications are multi-tiered (up to four levels) for increased reporting. A single data set can be part of different applications. For example, reports are available by user, project, department, and company. Application classification are based on DFSMS Storage Group, Management Class, Storage Class, Data Class, data set name, data set organization, application code level, volume, qualifier, or user-defined fields.

The SGCONTRL function reduces the required user input by using a dynamic application generator. Space usage is tracked dynamically and recorded in the application database to provide utilization totals and high-water marks. DASD is tracked for permanent, temporary, and VSAM data sets. Separate totals are also maintained for DFHSM migrate and backup data sets. High-water marks based on actual usage provide accurate trending and forecasting data.

System Parameters

Table 2-1 and Table 2-2 describe the system parameters that affect the SGCONTL function.

Table 2-1 System Parameters Specific to the SGCONTL Function

Parameter	Purpose
SGC_ADDEXIT	Defines the name of the Add Exit.
SGC_CHKEXIT	Defines the name of the Check Exit.
SGC_DEFEXIT	Defines the name of the Default Exit.
SGC_KEYEXIT	Defines the name of the Application Code Build Exit.
SGC_SECEXIT	Defines the name of the Security Exit.
SGC_SELEXIT	Defines the name of the Select Exit.
SGC_STOGRP	Specifies whether to retrieve SMS storage group information. This parameter should be set to YES only if SMS storage group information is required for FLIST or RLIST processing.
SGC_STORCLS	Specifies whether to retrieve class information. This parameter may be set to YES only if SMS class information is required for FLIST or RLIST processing.
SGCDSN	Defines the data set name of the application database. The DDNAME SGCDB is dynamically allocated to this name with startup of SVSGC when the SGCDB DD is not present in the SVOS start up JCL.
SGCSCALE	Specifies the default scaling value for budget space amounts.

Table 2-2 System Parameters Common to MAINVIEW SRM

Parameter	Purpose
AOO_SYBSYS	Specifies the AutoOPERATOR subsystems that are to receive events.
BBI3_SSID	Specifies the CAS subsystem name to which the SVOS PAS should connect.
DIAG	Specifies the suffix of parameter member SMDIAGxx.
DIAGMSDD	Establishes WTO message tracing.
DISPLAY	Displays functions list in the ISPF interface.
DUMPDD	Provides a SYS1.DUMPxx dump in the event of a MAINVIEW SRMabend.
EVNT	Specifies the suffix you assign to the name of the SMEVNTxx event definition member.
FORPLEXNAME	Specifies one or more user-defined systems that can be included or excluded in a sysplex environment.
FORSMFID	Specifies SMF records that can be included or excluded in a sysplex environment.
FORSYSID	Specifies user-defined system IDs that can be included or excluded in a sysplex environment.
FUNC	Specifies the suffix of parameter member SMFUNCxx.
IGNOREDD	Suppresses MAINVIEW SRM activity for a job step.
MODTRCDD	Sets module entry and exit tracing.

Table 2-2 System Parameters Common to MAINVIEW SRM

Parameter	Purpose
MSGID	Specifies the inclusion of the MAINVIEW SRM message identifier in the message text.
MSGLVL	Specifies the level of messages to be generated.
PASSWORD	Specifies the password(s) supplied by BMC Software.
POOL	Specifies the suffix of parameter member SMPOOLxx.
SMFID	Specifies the record number for MAINVIEW SRM SMF records.
SYSLIB	Specifies a default data set to be allocated at SVOS startup
SYSLIB2	Specifies a default data set to be allocated at SVOS startup
SYSLIB3	Specifies a default data set to be allocated at SVOS startup.
TRACEDD	Traces MAINVIEW SRM activity for jobstep.
VAR	Specifies the suffix of parameter member SMVARSxx
WTORDC	Specifies the message descriptor coed(s).
WTORC	Specifies the routing code to be assigned to the message text. For more information, see the WTO macro ROUTCDE= parameter in the <i>MVS Supervisor Services and Macro Instructions</i> .

Rule List Parameter Reference

Table 2-3 presents syntax and descriptions for the application collector rule list parameters (in member SMFLSTxx).

Table 2-3 Rule List Parameter Reference

Parameter	Description
EVENTID=xxxxx	Identifies a user event defined in an SMEVNTxx member.
NEWAPPL=xxxxxxxx	Specifies the value of an application code that is used to override the default application code. The NEWAPPL parameter replaces the NEWACCT parameter, which also remains valid.
SCAN=EXIT	Specifies not to budget space for any level associated with a data set; this parameter is unique to the application collector.

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

EVENTID=

Purpose: Specifies the identifier assigned to a user event in SMEVNTxx. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=xxxxx

where xxxxx is the 5-character string specified on the EVNTID parameter in SMEVNTxx.

Default: None

NEWAPPL=

Purpose: Specifies the value of an application code that is used to override the default application code. The NEWAPPL parameter replaces the NEWACCT parameter, which also remains valid.

Syntax: NEWAPPL=xxxxxxxx

where xxxxxxx is an alphanumeric code 1 to 50 characters long

When NEWAPPL=NONE is specified, no processing is done for the data set for the level(s) specified.

Default: None

SCAN=

Purpose: Specifies not to budget space for any level associated with a data set for the application collector.

Syntax: SCAN=EXIT

Supported Filter List Parameters

The following are the only filter list parameters that are available for the SGCONTRL function.

DATACLAS	DSNAME	DSNTYPE	DSORG
DSTYPE	HSM	HSMDSN	JOB
JOBACCT	JOBCLASS	JOBTYPE	LEVEL
LLQ	MGMTCLAS	OLDACCT	PGMRNAME
POOL	QUALL	QUALn	RACFGRP
RACFUID	SGC_FUNC	SMS	STORCLAS
STORGRP	SYSID	TEMPDSN	USRCn
VOLSER			

Usage Notes

Use the NEWAPPL statement to override the default application code.

Note: The system skips space budgeting on the current level if the new application code is blank. The new application code cannot exceed 50 bytes or the value specified for MAXKEY when the application database was created, whichever is less.

The syntax for defining a SET NEWAPPL= statement is

```
SET NEWAPPL='text' ,(keyword,displ,len,NOBLANK|BLANK),HIT
```

The application collector can substitute values into your new application code symbolically so you can build codes with information from various sources. Use the NEWAPPL statement operands to enable this feature.

'text'

Function: Contains the text of the replacement application code. The text can be 1 to 50 characters in length. The text must be enclosed in single quotation marks if it contains embedded spaces or commas.

Data can be substituted into text dynamically by specifying a *hit* character in the text. The default hit character is ?. The application collector scans the text for a question mark (?). Any question mark encountered is replaced with information in the next positional operand. Scanning for symbolic substitution continues until all question marks have been replaced. If the new text requires a question mark, the hit character can be modified with the HIT operand in the NEWAPPL statement. In the following example, the volume name is substituted into the new application code, concatenated to the literal SYSTEMS.

```
SET NEWAPPL=' SYSTEMS?' ,VOLSER
```

To substitute a substring of the operand following the hit character, enclose in parentheses the substitution text and two more operands that specify the substring displacement and length. The following example uses the first 3 bytes of the volume name.

```
SET NEWAPPL=' SYSTEMS?' , (VOLSER, 1, 3)
```

The new application code above is 10 characters long: the 7-character literal SYSTEMS followed by the first 3 characters of the VOLSER. For example, if the VOLSER is STR001, the NEWAPPL is SYSTEMSSTR.

To substitute variable information without literals, type one question mark for each substitution desired. In this example, the first-level qualifier will be suffixed by the last qualifier.

```
SET NEWAPPL='??' ,QUAL1 ,QUALL
```

The value of NEWAPPL can be any combination of literals and variable information, but it cannot exceed the length of the application code as defined with the MAXKEY parameter during initialization.

Default: None

Format: Any valid text string from 1 to 50 characters in length.

```
(keyword,displ,length,NOBLANK | BLANK) . . .
```

Function: A positional operand that specifies the dynamic substitution data used to dynamically build the replacement application code for the application.

You must specify as many of these positional operands as there are hit characters within the text. The keyword operand identifies the data to be inserted.

If you specify only the keyword, the entire string that it represents is used. You must use parentheses if you want a substring of the keyword. The displacement (displ) defaults to the first character and length defaults to the length associated with the keyword. The following are valid keywords:

QUAL1	QUAL2	QUAL3	QUAL4	QUAL5
QUAL6	QUAL7	QUAL8	QUALL	DSNAME
VOLSER	DSORG	USRC1	USRC2	TEMPDSN
LEVEL	OLDACCT	DATACLAS	STORCLAS	MGMTCLAS
STORGRP	SMS			

If you specify NOBLANK at the end of the operand, trailing blanks are removed from the string during substitution. The default (BLANK) does *not* remove blanks from the end of a string during substitution. Since these operands are positional, you must code a comma for those that you omit.

In the following example, the first- and second-level qualifiers are substituted into the application. Any blanks are removed.

```
SET NEWAPPL='??', (QUAL1, , , NOBLANK), (QUAL2, , , NOBLANK)
```

Performance Collector Functions

Table 2-4 provides a description of performance collector functions and lists valid filter list parameters for each. Table 2-5 on page 2-14 describes the filter list parameters. There are no valid rule list parameters for performance collector functions.

Table 2-4 Performance Collector Functions (Part 1 of 5)

Function	Description and Examples	Filter List Parameters
SGDACCT	application utilization thresholds; allows events to be generated from thresholds on values in the group utilization record.	EVENTID SGDA_ALNV SGDA_ALV SGDA_AVAIL SGDA_GRP SGDA_IDLE SGDA_NVDS SGDA_VDS
SGDPOOL	pool utilization thresholds; allows events to be generated from thresholds on values in the pool-utilization record.	EVENTID SGDP_ALNV SGDP_ALV SGDP_AVAIL SGDP_IDLE SGDP_NCLPER SGDP_NNV SGDP_NV SGDP_NVOL SGDP_POOL SGDP_RSVD SGDP_RVAARC SGDP_RVAFNC SGDP_RVAFSC SGDP_RVAIND SGDP_RVANCL SGDP_TYPE SGDP_PERFUL

Table 2-4 Performance Collector Functions (Part 2 of 5)

Function	Description and Examples	Filter List Parameters
SGDVOL	<p>volume utilization thresholds; allows events to be generated from thresholds on values in the volume utilization record.</p>	EVENTID SGDV_ALREXT SGDV_FRAGI SGDV_FRCYL SGDV_FREXT SGDV_FRVIR SGDV_IDTR SGDV_LREXT SGDV_LREXTT SGDV_NDS SGDV_NF0DSC SGDV_POOL SGDV_POOL1 SGDV_PTYP SGDV_RSRVDT SGDV_RVAFDV SGDV_RVAIND SGDV_RVAPCS SGDV_RVAPCU SGDV_RVASSF SGDV_RVAVOL SGDV_USEXT SGDV_VOL SGDV_PERFUL
SGPCCURC	<p>cache controller records; controls the inclusion or exclusion of the cache controller records.</p> <p>Use SGPCCURC to include only cache controllers in which the I/Os-per-second rate is greater than 10 or the read-hit percentage is less than 25.</p> <pre>SET MODE=ACT INC SGP_IOPRSEC>10 INC SGP_RDHIT@<25</pre>	EVENTID JOB SGP_CNTLUID SGP_IOPRSEC SGP_NRDHIT@ SGP_NRDPSEC SGP_NWRHIT@ SGP_NWRTPSC SGP_RDHIT@ SGP_RDSPRSC SGP_READ@ SGP_SRDHIT@ SGP_SRDPRSC SGP_SWRHIT@ SGP_SWRPRSC SGP_WRHIT@ SGP_WRITE@ SGP_WRPSEC

Table 2-4 Performance Collector Functions (Part 3 of 5)

Function	Description and Examples	Filter List Parameters
SGPCPREC	<p>channel path records; controls the inclusion or exclusion of the channel path records members.</p> <p>Use SGPCPREC to include only channel paths that are busy more than 25 percent of the time.</p> <pre>SET MODE=ACT INC SGP_@BUSY>25</pre>	EVENTID JOB SGP_CHPID SGP_DP@BUSY SGP_IOPRSEC
SGPDSREC	<p>data set records; controls the inclusion or exclusion of the data set records.</p> <p>Use SGPDSREC to exclude all data sets with a name that starts with SYSTEM and that reside on volume 111111.</p> <pre>SET MODE=ACT EXC DSN=SYSTEM/ VOL=111111</pre> <p>Include only data sets that have an I/Os-per-second rate that is greater than 30 or a response time greater than 10 milliseconds.</p> <pre>SET MODE ACT INC SGP_IOPRSEC>30 INC SGP_RESPTIM>100</pre> <p>The response time is stated in .1-millisecond units; therefore, 100 is 10 milliseconds.</p>	DSN DSTYPE EVENTID JOB SGP_CONNTIM SGP_DISCTIM SGP_IOPRSEC SGP_IOSQTIM SGP_JOB_CNT SGP_PENDTIM SGP_RDHIT@ SGP_READ@ SGP_RESPTIM SGP_SERVTIM SGP_WRHIT@ SGP_WRITE@ STORCLAS VOL
SGPFILTR	<p>data set type filter; controls the inclusion or exclusion of the data set record based on the data set type.</p> <p>SGPFILTR applies to all data sets. Filtering temporary data sets with SGPFILTR will give better performance than using the DSTYPE keyword in the SGPDSREC member.</p> <p>Use SGPFILTR to exclude all data sets whose name starts with SYSTEM and are GDGs.</p> <pre>SET MODE=ACT EXC DSN=SYSTEM/ DSTYPE=GDG</pre>	DSN DSTYPE EVENTID

Table 2-4 Performance Collector Functions (Part 4 of 5)

Function	Description and Examples	Filter List Parameters
SGPJBIRC	<p>job records; controls the inclusion or exclusion of the job records.</p> <p>Use SGPJBIRC to exclude all jobs whose name starts with SMF and have I/Os-per-second rate is less than 20.</p> <pre>SET MODE=ACT EXC JOB=SMF/SGP_IOPRSEC<20</pre>	EVENTID JOB SGP_CONNTIM SGP_DISCTIM SGP_IOPRSEC SGP_IOSQTIM SGP_PENDTIM SGP_RDHIT@ SGP_READ@ SGP_RESPTIM SGP_SERVTIM SGP_WRHIT@ SGP_WRITE@
SGPLCURC	<p>logical control unit records; controls the inclusion or exclusion of the logical control unit records.</p> <p>Use SGPLCURC to include only LCUs whose director port is busy more than 25 percent of the time, or the LCU itself is busy more than 50 percent of the time.</p> <pre>SET MODE=ACT INC SGP_DP@BUSY>25 INC SGP_LCU@BSY>50</pre>	EVENTID JOB SGP_DP@BUSY SGP_IOPRSEC SGP_LCUID SGP_LCU@BSY
SGPPSMRC	<p>storage pool records; controls the inclusion or exclusion of the storage pool records.</p> <p>Use SGPPSMRC to exclude all pools whose name starts with SAM and have I/Os-per-second rate is less than 20.</p> <pre>SET MODE=ACT EXC POOL=SAM/SGP_IOPRSEC<20</pre>	EVENTID JOB POOL SGP_ALLCSPC SGP_ALLOC@ SGP_IOPRSEC SGP_RESPTIM SGP_TOTSPAC
SGPRSFRC	<p>RVA subsystem frame record creation; controls processing for the IBM RAMAC Virtual Array (RVA) subsystem frame resource.</p> <p>To implement this functionality, you must first make JCL changes in the space collector.</p>	EVENTID SGP_BESCOLT SGP_BESFREE SGP_BESTOTL SGP_BESUNCL SGP_ECMCFBS SGP_ECMMSGs SGP_ECMNSPC SGP_ECMPGMS SGP_FSCBYRD SGP_FSCPERC SGP_FSUPERC SGP_NCL SGP_NCLPERC SGP_RSNAME

Table 2-4 Performance Collector Functions (Part 5 of 5)

Function	Description and Examples	Filter List Parameters
SGPSCLRC	<p>storage class summary records; controls the inclusion or exclusion of the storage class summary records.</p> <p>Use SGPSCLRC to exclude all storage classes whose name starts with CICS and have I/Os-per-second rate is less than 20. SET MODE=ACT EXC STORCLAS=CICS/ SGP_IOPRSEC<20</p>	EVENTID JOB SGP_CONNTIM SGP_DISCTIM SGP_DSNCNT SGP_IOPRSEC SGP_IOSQTIM SGP_PENDTIM SGP_RDHIT@ SGP_READ@ SGP_RESPTIM SGP_SERVTIM SGP_WRHIT@ SGP_WRITE@ STORCLAS
SGPVOLRC	<p>volume records; controls the inclusion or exclusion of the volume records.</p> <p>Use SGPVOLRC to exclude all volumes whose serial starts with WRK, or volumes whose I/Os-per-second rate is less than 20. SET MODE=ACT EXC VOL=WRK/ SGP_IOPRSEC<20</p>	EVENTID JOB SGP_@BUSY SGP_CFWHIT@ SGP_CFWPRSC SGP_CONNTIM SGP_CUBSYDL SGP_DFWHIT@ SGP_DFWPRSC SGP_DISCTIM SGP_DPBSYDL SGP_DSALLOC SGP_DVBSYDL SGP_IOPRSEC SGP_IOSQTIM SGP_NRDHIT@ SGP_NRDPRSC SGP_PENDTIM SGP_RDHIT@ SGP_READ@ SGP_RESERV@ SGP_RESPTIM SGP_SERVTIM SGP_SRDHIT@ SGP_SRDPRSC SGP_WRHIT@ SGP_WRITE@ STORGRP VOL

Table 2-5 lists all of the filter list parameters used by the performance collector and the space collector functions. Because parameters are used for more than one function, they are listed in alphabetical order in this table. Within the description of each function, there is a list of parameters used by that function.

Table 2-5 Filter List Parameters for Performance and Space Collectors (Part 1 of 4)

Parameter	Description
DSN=xxxxxxxxxxxxxxxx	data set name (1–44 characters)
DSTYPE=xxxx	data set type (PERM, TEMP, GDG)
EVENTID=xxxxx	specifies an event identifier
JOB=xxxxxxxx	job, TSO, or STC name (1–8 characters)
SGDA_ALNV=nnnnnnnnnn	specifies the total space allocated to non-VSAM data sets in the account
SGDA_ALV=nnnnnnnnnn	specifies the total space allocated to VSAM data sets in the account
SGDA_AVAIL=nnnnnnnnnn	specifies the total space available in the account
SGDA_GRP=xxxxxxx	specifies the application collector group name; also known as account name
SGDA_IDLE=nnnnnnnnnn	specifies the total allocated space that is unused in the account
SGDA_NVDS=nnnnn	specifies the number non-VSAM data sets in the account
SGDA_VSD=nnnnn	specifies total number of VSAM data sets in the account
SGDP_ALNV=nnnnnnnnnn	specifies the space allocated for non-VSAM data sets in the pool
SGDP_ALV=nnnnnnnnnn	specifies the total space allocated to VSAM data sets in the pool
SGDP_AVAIL=nnnnnnnnnn	specifies the total space available in the pool
SGDP_IDLE=nnnnnnnnnn	specifies the space allocated and unused in the pool
SGDP_NCLPER=nnnn	specifies the net capacity load percentage in tenths of a percent (0-1000)
SGDP_NNV=nnnnn	specifies the number of non-VSAM data sets in the pool
SGDP_NV=nnnnn	specifies the number of VSAM data sets in the pool
SGDP_NVOL=nnnnn	specifies the number of volumes in the pool
SGDP_PERFUL	specifies the Percentage Full or Percentage Allocated for all volumes in the pool
SGDP_POOL=xxxxxxx	specifies the pool name for reporting
SGDP_RSVD=nnnnnnnnnn	specifies the total reserved space in the pool
SGDP_RVAARC=nnnnnnnnnn	specifies the array capacity of the device for RVA pools
SGDP_RVAIND=YES/NO	specifies whether the pool is for an RVA device
SGDP_RVANCL=nnnnnnnnnn	specifies the net capacity load of the RVA device
SGDP_RVAFSC=nnnnnnnnnn	specifies the amount of space collected by free space collection activity during the interval for RVA pools
SGDP_RVAFNC=nnnnnnnnnn	specifies the amount of space not collected by free space collection activity during the interval for RVA pools

Table 2-5 Filter List Parameters for Performance and Space Collectors (Part 2 of 4)

Parameter	Description
SGDP_TYPE= <i>x</i>	specifies the type of pool
SGDV_ALREXT= <i>nnnnn</i>	specifies the number of additional tracks in largest free extent on the volume
SGDV_FRAGI= <i>nnnnn</i>	specifies the fragmentation index on the volume
SGDV_FRCYL= <i>nnnnn</i>	specifies the number of free cylinders on the volume
SGDV_FREXT= <i>nnnnn</i>	specifies the number of free extents on the volume
SGDV_FRVIR= <i>nnnnn</i>	specifies the free VIR count on the volume
SGDV_IDTR= <i>nnnnn</i>	specifies the total number of idle tracks on the volume
SGDV_LREXT= <i>nnnnn</i>	specifies the number of cylinders in largest free extent on the volume
SGDV_LREXTT= <i>nnnnn</i>	specifies the size of largest extent in tracks on the volume
SGDV_NDS= <i>nnnnn</i>	specifies the total number of data sets on the volume
SGDV_NF0DSC= <i>nnnnn</i>	specifies the format 0 (free) DSCB count on the volume
SGDV_PERFUL	specifies the Percentage Full or Percentage Allocated for the volume
SGDV_POOL= <i>xxxxxxxx</i>	specifies the first pool name in which the volume is defined
SGDV_POOL1= <i>xxxxxxxx</i>	specifies pool name in which the volume is defined
SGDV_PTYP= <i>x</i>	specifies the pool type
SGDV_RVAIND= <i>YES/NO</i>	indicates whether the volume exists on a RVA frame
SGDV_RVAFDV= <i>xxxxxxxx</i>	specifies the functional device ID for a volume existing on a RVA frame
SGDV_RVAPCS= <i>nnnnn</i>	specifies the physical capacity shared for a volume existing on a RVA device
SGDV_RVAPCU= <i>nnnnn</i>	specifies the physical capacity used for a volume existing on a RVA device
SGDV_RVASSF= <i>xxxxxxxx</i>	specifies the RVA subsystem frame name for the RVA frame the volume exists on
SGDV_RVAVOL= <i>xxxxxxxx</i>	specifies the descriptive volume name of a volume existing on a RVA frame
SGDV_RSRVDT= <i>nnnnn</i>	specifies the number of reserved tracks (not included in free space) on the volume
SGDV_USEXT= <i>nnnnn</i>	specifies the number of used extents on the volume
SGDV_VOL= <i>xxxxxxxx</i>	specifies the volume serial number of the volume
SGP_@BUSY=>< <i>nnn</i>	specifies channel path busy threshold for inclusion or exclusion
SGP_BESCOLT= <i>nnnnnnnn</i>	specifies the collected back-end space in tenths of a MB
SGP_BESFREE= <i>nnnnnnnn</i>	specifies the free back-end space in tenths of a MB
SGP_BESTOTL= <i>nnnnnnnn</i>	specifies the total back-end space in tenths of a MB
SGP_BESUNCL= <i>nnnnnnnn</i>	specifies the uncollected back-end space in tenths of a MB
SGP_CFWHIT@=>< <i>nnn</i>	specifies number of CFAST writes reads per-second threshold
SGP_CFWPRSC=>< <i>nnn</i>	specifies percentage of CFAST writes satisfied by cache threshold
SGP_CHPID=>< <i>xx</i>	specifies channel paths to be included or excluded

Table 2-5 Filter List Parameters for Performance and Space Collectors (Part 3 of 4)

Parameter	Description
SGP_CNTLUID=><xx	specifies subsystem IDs of cache controllers to be included or excluded
SGP_CONNTIM=><nnnnn	specifies the data set connect time threshold in .1 millisecond increments
SGP_CUBSYDL=><nnnnn	specifies the control unit busy delay threshold in .1 millisecond increment
SGP_DFWRIT@=><nn	specifies percentage of DFAST writes satisfied by cache threshold
SGP_DFWRPSC=><nnn	specifies number of DFAST writes per-second threshold
SGP_DISCTIM=><nnnnn	specifies the data set disconnect time threshold in .1 millisecond increments
SGP_DP@BUSY=><nn	specifies director port busy percentage to be included or excluded
SGP_DPBSYDL=><nnnnn	specifies the director port busy delay time threshold in .1 millisecond increments
SGP_DVBSYDL=><nnnnn	specifies the device busy delay time threshold in .1 millisecond increments
SGP_ECMCFBS=nnnnnnnn	specifies the ECAM channel programs bypassed due to busy configuration in tenths of a MB
SGP_ECMMSGS=nnnnnnnn	specifies ECAM messages processed in tenths of a MB
SGP_ECMNSPC=nnnnnnnn	specifies the ECAM channels programs bypassed due to no buffer space in tenths of a MB
SGP_ECMPGMS=nnnnnnnn	specifies the ECAM channel programs in tenths of a MB
SGP_FSCBYRD=nnnnnnnn	specifies the collected free space bytes read in tenths of a MB
SGP_FSCPERC=nnnn	specifies the percentage of collected free space in tenths of a percent
SGP_FSUPERC=nnnn	specifies the net capacity load percentage in tenths of a percent
SGP_IOPRSEC=><nnn	specifies number of IOs per-second threshold
SGP_IOSQTIM=><nnnnn	specifies the data set IOSQ time threshold in .1 millisecond increments
SGP_LCU@BUSY=><nn	specifies LCU busy percentage to be included or excluded
SGP_LCUID=><xx	specifies the logical control unit id of those controllers to be included or excluded
SGP_NCLPERC=nnnn	specifies the percentage of uncollected free space in tenths of a percent
SGP_NRDHIT@=><nnn	specifies percentage of normal reads satisfied by cache threshold
SGP_NRDPSEC=><nnn	specifies number of normal reads per-second threshold
SGP_NWRHIT@=><nnn	specifies percentage of normal writes satisfied by cache threshold
SGP_NWRTPSC=><nnn	specifies number of normal writes per-second threshold
SGP_PENDTIM=><nnnnn	specifies the data set pending time threshold in .1 millisecond increments
SGP_RDHIT@=><nnn	specifies percentage of reads satisfied by cache threshold
SGP_RDSPRSC=><nnn	specifies number of reads per-second threshold
SGP_READ@=><nnn	specifies the percentage of IOs that are reads threshold
SGP_RESERV@=><nn	specifies percentage volume is reserved for inclusion or exclusion
SGP_RESPTIM=><nnnnn	specifies the data set response time threshold in .1 millisecond increments
SGP_RSFNAME=xxxxxxx	specifies the IXP subsystem frame name

Table 2-5 Filter List Parameters for Performance and Space Collectors (Part 4 of 4)

Parameter	Description
SGP_SRDHIT@=><nnn	specifies percentage of sequential reads satisfied by cache threshold
SGP_SRDPRSC=><nnn	specifies number of sequential reads per-second threshold
SGP_SWRHIT@=><nnn	specifies percentage of sequential writes satisfied by cache threshold
SGP_SWRPRSC=><nnn	specifies number of sequential writes reads per-second threshold
SGP_WRHIT@=><nnn	specifies percentage of writes satisfied by cache threshold
SGP_WRITE@=><nnn	specifies percentage of IOs that are writes threshold
SGP_WRPRSEC=><nnn	specifies number of writes per-second threshold
VOL=xxxxxx	volume name (1–6 characters)

HSM Collector Functions

DFHSM reporting functions enable you to achieve the following objectives:

- control DFHSM backups
- enhance DFHSM deletion
- apply calendar conversion to DFHSM migration
- enhance DFHSM migration
- pool data sets during DFHSM recall
- control data set batches

DFHSM functions do not operate for DFDSS jobs.

HSMBACKP—Control DFHSM Backups

The HSMBACKP function enables you to select and exclude data sets and volumes for DFHSM backup.

DFHSM provides automated data set backup processing for OS/390. However, DFHSM does not provide an easy method of selecting groups of data sets for backup or for excluding groups of data sets from backup. Data sets must be included or excluded for backup processing either through aggregate group definitions or one data set at a time using the ALTERDS command.

HSMBACKP provides a simple and flexible means of selecting groups of data sets to be backed up by DFHSM or excluding groups of data sets from backup processing.

Table 2-6 provides a quick reference to parameters for the SET statement (in member SMRLSTxx). The section following the table explains each parameter.

Table 2-6 Parameter Quick Reference

Parameter	Description
BACKCMD= <u>YES/NO</u>	specifies whether command-initiated backups are processed
BACKUP= <u>YES/NO</u>	specifies that a backup will not be performed
EVENTID=xxxxx	identifies a user event that is defined in an SMEVNTxx member

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

BACKCMD=

Purpose: Specifies whether command-initiated backups are processed by HSMBACKP. BACKCMD=YES allows HSMBACKP to process command-initiated backups as well as automatic backups. BACKCMD=NO prohibits HSMBACKP from processing command-initiated backups. This is an optional parameter.

Syntax: BACKCMD=*YES/NO*

Default: BACKCMD=YES

BACKUP=

Purpose: Specifies whether the backup is allowed. If BACKUP=NO, DFHSM does not back up the selected resource. If BACKUP=YES, DFHSM backs up the resource. This is an optional parameter.

Syntax: BACKUP=*YES/NO*

Default: BACKUP=NO

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNTxx member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=xxxxx

The variable xxxxx is the 5-character string specified on the EVNTID parameter in the SMEVNTxx member.

Default: None

Usage Notes

HSMBACKP applies to all data sets selected by DFHSM for backup during volume backup or space management. It does not apply to data sets that were backed up explicitly, such as with the TSO/ISPF HBACKDS command.

The BACKCMD parameter enables you to bypass function processing for DFHSM backups that are initiated by command. HSMBACKP always processes automatically initiated backups, but it may be set to ignore backups that are performed on command. This enables you to perform emergency backups without modifying the normal backup parameters in HSMBACKP.

Note that the default mode of operation of HSMBACKP is to prohibit backups. The default is BACKUP=NO, which prevents a backup on the selected resource. Therefore, if most resources should be backed up unless otherwise specified, the rule list should conclude with BACKUP=YES for all resources (INC DSN=/).

When a backup is prohibited, HSMBACKP issues the following message:

```
SVM0380I dsn DISABLED FOR BACKUP
```

No message is issued when a resource is selected for backup.

Note that the HSMBACKP function must be specified in the SMFUNCxx member to be available and must be set to active to provide service.

The following example shows syntax for the HSMBACKP function:

SMFUNC3A member

SET	NAME=HSMBACKP	The function is defined and activated.
	ACTIVE=YES	
	MSG=I SMF=I	
	FLST=A5 RLST=A5	
	DESC='CONTROL BACKUPS'	

SMFLSTA5 member

SET	MODE=ACT	Select all data sets for processing.
	INC DSN=/	

SMRLSTA5 member

SET	BACKCMD=NO BACKUP=NO	Disallow automatic backups of test data sets. (Command-initiated backups are not affected.)
	INC DSN=**.TEST*	
SET	BACKCMD=NO BACKUP=NO	Do not back up transitory data sets. (Command-initiated backups are not affected.)
	INC DSN=**.TRANS*	
SET	BACKCMD=YES BACKUP=NO	Disallow all DFHSM backups against DB2® volumes.
	INC VOL=DB*	
SET	BACKCMD=YES BACKUP=YES	For all other data sets, allow automatic and command-initiated backups.
	INC DSN=/	

HSMDELETE—Enhance DFHSM Deletion

The HSMDELETE function provides enhanced processing when selecting DFHSM data sets for deletion.

HSMDELETE provides the following advantages over DFHSM processing:

- Aging can be based on working days, rather than calendar days.
- Data sets can be selected or excluded based on more criteria.

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNTxx member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=xxxxx

The variable xxxxx is the 5-character string specified on the EVNTID parameter in the SMEVNTxx member.

Default: None

Supported Filter List Parameters

Due to the invocation of HSMDELETE during OS/390 exit ARCSAEXT, only the following MAINVIEW SRM filter list parameters are available:

CALAGE	CAT	DSN	DSNn
DSORG	DSTYPE	GDGVER	HLQ
LLQ	POOL	RACF	REFAGE
SIZE	USER	VOL	XMODE

Usage Notes

HSMDELETE applies to all data sets considered by DFHSM for volume migration and space management migration.

If a data set is not selected for HSMDELETE processing by filter list specifications and the age of the data set is less than the age specified in the management class for data set deletion, HSMDELETE does not affect the processing of the data set.

The REFAGE and CALAGE selection parameters can be used to filter data sets considered by DFHSM for deletion. CALAGE contains the number of working days since the data set was created, while REFAGE contains the unadjusted number of days since the data set was created.

HSMDELETE issues the following message if it modifies the deletion status of a data set:

```
SVM0365I dsn DISABLED FOR AUTO DELETION
```

No message is issued when a data set is not selected by HSMDELETE.

Note that the HSMDELETE function must be specified in the SMFUNCxx member to be available and must be set to active to provide service.

The following example shows syntax for the HSMDELETE function.

SMFUNC12 member

```
SET    NAME=HSMDELETE ACTIVE=YES           The function is defined and
      FLST=09                               activated.
      MSG=I SMF=I
      DESC='CONTROL DELETION'
```

SMFLST09 member

```
SET    MODE=ACT
      INC DSN=TEST./ CALAGE>45             Select TEST. data sets if
      INC DSN=PROD./ CALAGE>90             CALAGE is greater than 45.
                                           Select PROD. data sets if
                                           CALAGE is greater than 90.
```

HSMCCNV—Apply Calendar Conversion to DFHSM Migration

The HSMCCNV function allows conversion of DFHSM migration periods for management classes based on the non-working day specifications in the MAINVIEW SRM calendar.

DFHSM's automatic space management processing selects data sets for automatic migration based in part on the number of days since the data set was last referenced (PRIMARY DAYS NON USAGE). This automatic migration is a valuable part of storage management. It requires little or no human intervention in moving low-usage data sets from high-speed devices to less expensive devices. However, DFHSM counts only elapsed calendar days in determining the time to migration; it does not take into account days that are not working days, such as public holidays, weekends, and installation-defined days off. This capability is provided by MAINVIEW SRM calendar definitions in member SMCALSxx.

During migration processing by DFHSM, HSMCCNV adjusts the PRIMARY DAYS NON USAGE specification by the number of non-working days in the period, thus giving an accurate elapsed time for migration based on actual workdays, rather than calendar days.

Table 2-7 provides a quick reference to parameters for the SET statement (in member SMRLSTxx). The section following the table explains each parameter.

Table 2-7 **Parameter Quick Reference**

Parameter	Description
CAL= <i>YES/NO</i>	specifies date conversion by calendar
EVENTID=xxxxx	identifies a user event defined in an SMEVNTxx member

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

CAL=

Purpose: Specifies whether the DFHSM management class migration days value should be adjusted to account for the non-working days in the MAINVIEW SRM calendar.

Syntax: CAL=*YES/NO*

Default: CAL=YES

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNTxx member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=xxxxx

The variable xxxxx is the 5-character string specified on the EVNTID parameter in the SMEVNTxx member.

Default: None

Supported Filter List Parameters

Filter and rule list entries for HSMMCCNV should include and exclude resources based only on the MGMTCLAS parameter.

Warning! The only valid selection parameter during HSMMCCNV processing is MGMTCLAS. All other selection parameters do not have a value when HSMMCCNV is active. Therefore, filter and rule lists used with HSMMCCNV should only INCLUDE or EXCLUDE based on MGMTCLAS. For example, the following filter list entry for HSMMCCNV will never be satisfied because the JOB parameter does not contain a value:

```
SET MODE=SIM
INC JOB=DFHSM/
```

This rule list entry will also never be satisfied because the VOL parameter does not contain a value.

```
SET CAL=NO
INC VOL=EMP/
```

Usage Notes

HSMGCCNV provides services only if CAL=YES is specified. HSMGCCNV ignores any management classes for which PRIMARY DAYS NON USAGE is zero.

The use-days figure is adjusted to workdays by starting at the current date and backing up for the number of workdays corresponding to use-days, adjusting the day count as non-working days are found in the calendar. This work-day count is returned to DFHSM as the PRIMARY DAYS NON USAGE value, which must be satisfied before a data set is migrated.

Note that the effect is to *increase* the primary days non-usage, so that the data set is migrated only after the number of actual working days has elapsed. Each non-working day in the period covered by the primary days non-usage adds a day to the number returned to DFHSM as the new primary days non-usage.

Note that the HSMGCCNV function must be specified in the SMFUNCxx member to be available and must be set to active to provide services.

HSMGCCNV issues the following message when it processes a management class:

```
SVM0361I  MGMTCLAS=xxxxxxxx, MD=nnnn, WD=nnnn
```

In this message, the variables are as follows:

- MGMTCLAS=xxxxxxxx indicates the management class name
- MD=nnnn identifies the original migration days value
- WD=nnnn identifies the adjusted migration days (working days) value

The following example shows syntax for the HSMGCCNV function:

SMFUNC21 member

```
SET      NAME=HSMGCCNV ACTIVE=YES  The function is defined and
MSG=E   SMF=N                      activated. SMF records are not
FLST=02  RLST=02                   written. Only error messages are
DESC='ADJUST MIGRATION DAYS'       written.
```

SMCAL10 member

SET YEAR=2000
 SAT=W
 SUN=F
 FREE=01.12 /* Dec 1 */
 FREE=01.01 /* Jan 1 */
 FREE=22.02 /* Feb 22 */
 FREE=17.04-20.04 /* Apr 17-20 */
 FREE=31.05 /* May 31 */
 FREE=04.07 /* July 4 */
 FREE=08.06 /* June 8 */
 FREE=25.11-26.11 /* Nov 25-26 */
 FREE=24.12 /* Dec 24 */
 FREE=25.12 /* Dec 25 */

The year 2000 is defined with Saturdays as work days, Sundays as non-working days, and various other days during the year non-working. (Note that it is not necessary to specify Saturday as a working day; any day or date not defined as FREE is a working day.)

SMFLST02 member

SET MODE=ACT
 INC MGMTCLAS=/

This filter list specification is active. It selects all resources with a management class.

SMRLST02 member

SET CAL=YES
 EXC MGMTCLAS=BATCH

This rule list specifies calendar adjustment for all management classes except BATCH.

HSM MIGRT—Enhance DFHSM Migration

The HSM MIGRT function provides enhanced processing when selecting DFHSM data sets for migration. HSM MIGRT provides the following advantages over DFHSM processing:

- Aging can be based on working days, rather than calendar days.
- Migrate data sets from ML0 to ML2 without a backup copy.
- Data sets can be selected or excluded based on more criteria.

Table 2-8 provides a quick reference to parameters for the SET statement (in member SMRLSTxx). The section following the table explains each parameter.

Table 2-8 Parameter Quick Reference

Parameter	Description
BACKUP= <u>YES/NO</u>	specifies that ML0-2 migration without backup is allowed
CAL= <u>YES/NO</u>	specifies date conversion by calendar
EVENTID=xxxxx	identifies a user event defined in an SMEVNTxx member
MIGCMD= <u>YES/NO</u>	specifies if command-initiated migration is processed
MIGRATE= <u>YES/NO</u>	specifies if migration is disallowed
ML2= <u>YES/NO</u>	specifies direct migration from ML0 to ML2

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

BACKUP=

Purpose: Specifies whether to allow migration from ML0 to ML2 without a backup copy is allowed. BACKUP=NO specifies that a backup copy is *not* necessary before ML0-ML2 migration. BACKUP=YES specifies that a backup copy must exist before migration from ML0 to ML2 is allowed. (See the ML2 parameter below.) This is an optional parameter.

Syntax: BACKUP=*YES/NO*

Default: BACKUP=YES

CAL=

Purpose: Specifies whether the DFHSM management class migration days value should be adjusted to account for the non-working days in the MAINVIEW SRM calendar.

Syntax: CAL=*YES/NO*

Default: CAL=YES

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNTxx member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=*xxxxx*

The variable *xxxxx* is the 5-character string specified on the EVNTID parameter in the SMEVNTxx member.

Default: None

MIGCMD=

Purpose: Specifies whether command-initiated migration is processed by HSMMIGRT. MIGCMD=YES allows HSMMIGRT to process command-initiated migration as well as automatic migration. MIGCMD=NO prohibits HSMMIGRT from processing command-initiated migration (that is, command-initiated migration overrides HSMMIGRT). This is an optional parameter.

Syntax: MIGCMD=*YES/NO*

Default: MIGCMD=YES

MIGRATE=

Purpose: Specifies whether migration is allowed. MIGRATE=YES allows migration processing to proceed for a data set; MIGRATE=NO disallows any migration of the data set. This is an optional parameter.

Syntax: MIGRATE=*YES/NO*

Default: MIGRATE=YES

ML2=

Purpose: Specifies whether direct migration from ML0 to ML2 is allowed. ML2=YES allows direct migration from ML0 to ML2; ML2=NO does not. This is an optional parameter.

Syntax: ML2=YES/NO

Default: ML2=NO

Supported Filter List Parameters

Due to the invocation of HSM MIGRT during OS/390 exits ARCSAEXT and ARCMDEXT, only the following MAINVIEW SRM filter list parameters are available:

CALAGE	CAT	DSN	DSNn
DSORG	DSTYPE	GDGVER	HLQ
LLQ	POOL	RACF	REFAGE
SIZE	USER	VOL	XMODE

Usage Notes

HSM MIGRT applies to all data sets considered by DFHSM for volume migration and space management migration. It does not apply to data sets migrated explicitly, such as with the DFHSM MIGRATE DSNAME command or the TSO or ISPF HMIGRATE command.

If a data set is not selected for HSM MIGRT processing by filter list or rule list specifications, DFHSM is directed to migrate the data set.

The REFAGE and CALAGE selection parameters can be used to filter data sets considered by DFHSM for migration. CALAGE contains the number of unreferenced working days, while REFAGE contains the unadjusted number of unreferenced days.

HSM MIGRT allows migration directly from ML0 to ML2 with or without a backup. Use the ML2 and BACKUP parameters to specify this service.

The MIGCMD parameter allows command-initiated migration to bypass any selection or restriction by HSM MIGRT, ensuring that emergency migration requests proceed unhindered.

Note: If SETMIG NOMIGRATION or MIGRATION is set in DFHSM for a group of data sets or volumes, the MAINVIEW SRM exit will not be involved when these data sets are migrated.

Messages

HSM MIGRT issues one of the following messages if it modifies the migration status of a data set.

```
SVM0360I   dsn WILL MIGRATE TO ML1 MD=nnnn AGE=nnnn
           ML2 IF BACKUP EXISTS
           ML2 BYPASSING BACKUP
```

HSM MIGRT has modified the migration status of a data set as stated. MD is the DFHSM adjusted migration age. AGE=nnnn is the real age of the data set (elapsed days since last referenced).

```
SVM0362I
SVM0363I   dsn MADE INELIGIBLE FOR MIGRATION -MIGCMD
```

HSM MIGRT has modified the migration status of a data set to prevent its migration. MIGCMD indicates the rule directed no migration.

Note that HSM MIGRT could issue a large volume of messages. You may wish to suppress informational messages in the function definition in the SMFUNCxx parmlib member or in the filter list for the function.

No message is issued when a data set is not selected by HSM MIGRT.

Note that the HSM MIGRT function must be specified in the SMFUNCxx member to be available and must be set to active to provide service.

The following example shows syntax for the HSM MIGRT function.

SMFUNC12 member

```
SET  NAME=HSMIGRT ACTIVE=YES           The function is defined
     MSG=I SMF=I                       and activated.
     FLST=09 RLST=09
     DESC='CONTROL MIGRATION'
```

SMFLST09 member

```
SET  MODE=ACT                          Select all data sets for
     INC DSN=/                          processing.
```

SMRLST09 member

SET	MIGRATE=NO MIGCMD=NO INC SIZE<1MB INC HLQ=SYS*	Do not migrate data sets smaller than 1 megabyte or system data sets (but allow a command-initiated migration to override).
SET	MIGRATE=YES ML2=YES INC LLQ=LIST* DSORG=PS REFAGE=7 INC POOL=TEST REFAGE=7	Migrate listing data sets and data sets in the TEST pool directly to tape after seven calendar days of nonuse.
SET	MIGRATE=YES MIGCMD=NO INC DSORG=PO REFAGE=90	Migrate unused PDSs after 90 calendar days.
SET	MIGRATE=YES INC DSORG=VS CALAGE=120	Migrate unused VSAM data sets after 120 working days.
SET	MIGRATE=YES INC DSORG=PS SIZE>1MB SIZE<10MB X CALAGE=45	Migrate unused sequential data sets between 1 and 10 MB in size after 45 working days.
SET	MIGRATE=YES INC DSORG=PS SIZE>10MB CALAGE=30	Migrate unused sequential data sets larger than 10 MB after 30 working days.

HSMRECAL—Pool Data Sets During DFHSM Recall

The HSMRECAL function provides DASD pooling support for non-DFSMS data sets recalled by DFHSM.

If a migrated data set is needed for processing, DFHSM recalls it from the migration volume to a DASD volume. DFHSM selects destination volumes for non-DFSMS-managed data sets based on maximum available space within a group of volumes selected on the basis of data set pool, original volume pool, use attribute, defined recall attributes, and so on.

HSMRECAL allows recalled data sets to be assigned to MAINVIEW SRM-defined pools on the basis of data set attributes including name, organization, size, and so on. Within eligible pools, volumes are selected based on the best fit of the data set size to available extents.

Table 2-9 provides a quick reference to parameters for the SET statement (in member SMRLSTxx). The section following the table explains each parameter.

Table 2-9 Parameter Quick Reference

Parameter	Description
EVENTID=xxxxx	identifies a user event defined in an SMEVNTxx member
POOL=(xxxxxx, . . .)	1-15 pool names from (SMPOOLxx)
USEVOL=STOR/PRI V/ALL	selects volume by use attribute (storage and/or private)

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNTxx member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID=xxxxx

The variable xxxxx is the 5-character string specified on the EVNTID parameter in the SMEVNTxx member.

Default: None

POOL=

Purpose: Specifies the name of the pool to receive the recalled data set. If multiple pools are specified, the first pool with sufficient space for the data set is assigned. This is a required parameter.

Syntax: POOL=(xxxxxx, xxxxxx, . . .)

The variable xxxxxx is a 1-8 character pool name defined in SMPOOLxx. If the pool name is not defined, refresh or start-up will fail. Up to 15 pools can be specified. If a single pool is specified, parentheses are not needed.

Default: None

USEVOL=

Purpose: Specifies the type of volume (storage, private, or any) that satisfies the allocation. USEVOL=STOR directs the data set to a storage-mounted volume; USEVOL=PRIV directs the data set to a private volume. (There is currently no specification for PUBLIC-only volumes.) USEVOL=ALL directs the allocation to the first available volumes, regardless of use attribute. This is an optional parameter.

Syntax: USEVOL=xxx

The variable xxx represents one of the following types of volumes:

STOR Storage
 PRIV Private
 ALL All volumes, regardless of use attribute

Default: USEVOL=ALL

Supported Filter List Parameters

Due to the invocation of HSMRECAL during OS/390 exit ARCRDEXT, only the following MAINVIEW SRM filter list parameters are available:

CAT	DEVTYPE	DSN	DSNn
DSORG	HLQ	JOB	JOBACCTn
JOBCLASS	LLQ	PGM	PGMRNAME
PROCSTEP	RACF	SIZE	STEP
STEPACCTn	SYSID	USER	VOL
XMODE			

Usage Notes

HSMRECAL applies to migrated data sets being recalled by DFHSM from ML1 or ML2 volumes. HSMRECAL does not process DFSMS-managed data sets.

DFHSM passes a list of candidate destination volumes to HSMRECAL. HSMRECAL returns up to four volumes that are in the candidate list and that are also in one or more of the specified MAINVIEW SRM pools. If HSMRECAL cannot find any of the DFHSM candidate volumes in any of the specified pools, the function makes no volume recommendations, thus allowing DFHSM to select the volume.

HSMRECAL requires that all pool volumes that are eligible for recall be defined to DFHSM with the AUTORECALL option on the ADDVOL command. Also, the SETSYS RECALL command must specify PRIVATEVOLUME(UNLIKE).

Note that parameters related to job execution (such as JOB, JOBCLASS, PGM, and so on) will have values for DFHSM. DASDPOOL rules based on such parameters may not give the desired results during HSMRECAL.

Note that the HSMRECAL function must be specified in the SMFUNCxx member to be available and must be set to active to provide service.

Messages

HSMRECAL issues the following message when a destination pool is selected for the recalled data set:

```
SVM0370I  dsn POOL (poolname) ASSIGNED
```

Note: If the SVM0370I message contains the pool name \$DHSSM\$, HSMRECAL is unable to pool the data set being recalled. HSM will chose the volume on which the data set will be recalled. This can occur because either of the following is true:

- You have coded a pooling rule that attempts to pool the data set to a volume that is not HSM-managed.
- The volume from which the data set was migrated is defined as part of a VOLUMEPOOL in the HSM command file (SYS.PARMLIB(ARCCMDxx)).

The following example shows syntax for the HSMRECAL function.

SMFUNC05 member

SET NAME=HSMRECAL ACTIVE=YES The function is defined and activated.
 MSG=I SMF=I
 FLST=5B RLST=5F
 DESC='CONTROL RECALL ALLOC'

SMPOOL11 member

SET POOLNAME=STD USELIMIT=90 Pool PRODSTD includes all volumes whose names begin
 VOL=PROD0/ with PROD0. A 90% capacity limit is put on all volumes in
 this pool.

SET POOLNAME=PRODMAX Pool PRODMAX includes two high-capacity 3390 volumes
 USELIMIT=80 designated for very large data sets.
 VOL=PROD2/

SET POOLNAME=TEST00 Pool TEST00 includes all volumes whose names begin with
 EXC VOL=TESTC/ TEST, except those beginning with TESTC.
 INC VOL=TEST/

SET POOLNAME=WORK Pool WORK includes the two named volumes.
 VOL=(WORK01,WORK02)

SMFLST5B member

SET MODE=ACT Allow DFHSM to process VSAM master files; all other data
 EXC LLQ=MSTR* DSORG=VS set recalls go through HSMRECAL.
 INC DSN=/

SMRLST5F member

SET POOL=WORK Data sets smaller than 1 megabyte are assigned to pool
 INC SIZE<1MB WORK.

SET POOL=(TEST00,WORK) Test data sets and data sets belonging to DEV are assigned
 INC LLQ=*TST* to either pool TEST00 or pool WORK.
 INC RACF=DEV

SET POOL=TEST00 Data sets larger than 1 megabyte and smaller than 10
 INC SIZE>1MB SIZE<10MB megabytes are assigned to pool TEST00.

SET POOL=PRODMAX Data sets larger than 50 megabytes and VSAM data sets
 INC SIZE>50MB larger than 10 megabytes are assigned to pool PRODMAX.
 INC SIZE>10MB DSORG=VS

SET POOL=STD All other data sets are assigned to pool STD.
 INC DSN=/

SMHSMUTL—DFHSM Control Data Set Batch Utility

The SMHSMUTL batch program purges obsolete entries based on user-specified criteria from the DFHSM migration and backup control data sets.

SMHSMUTL reads the DFHSM control data sets, selects data set entries on the basis of your specifications and generates TSO DFHSM commands to delete migration and backup entries and associated data sets.

You can select data sets for deletion based on data set name (full name or name mask), MAINVIEW SRM pool membership, or any of four aging criteria, with or without non-working day adjustment by the MAINVIEW SRM calendar.

Table 2-10 provides a quick reference to parameters for the SET statement (in member SMRLSTxx). The section following the table explains each parameter.

Table 2-10 Parameter Quick Reference

Parameter	Description
DELETE	keyword parameter that must precede all other parameters
BAKDAY=nnnn	age at which backup entries can be deleted
CAL=YES/NO	specifies use of MAINVIEW SRM calendar adjustment services
CDS=MCDS/BCDS/BOTH	type of control data set (migration, backup, or both)
CREDDAYS=nnnn	age (from creation date) at which migration entries can be deleted
DSN=xxxxxxxx. . .	name or name mask of data set(s) to be selected
EVENTID=xxxxx	identifies a user event defined in an SMEVNTxx member
KEEPBACKV=nnnn	number of copies of backup data sets that should be kept
MIGDDAYS=nnnn	age (from migration date) at which migration entries can be deleted
POOL=xxxxxxxx	a single MAINVIEW SRM pool name
REFDDAYS=nnnn	age (from last reference) at which migration entries can be deleted

Parameter Explanations

This section defines the purpose, syntax, and default value of each parameter.

DELETE=

Purpose: Identifies a new specification. This must be the first operand on a control card set for SMHSMUTL. Each DELETE keyword starts a new specification.

Syntax: DELETE

Default: None

BAKDAY=

Purpose: Specifies the age (number of days since last backup) at which the backup data set (and BCDS entry) is deleted. This is a required parameter if CDS=BCDS. If CDS=MCDS or CDS=BOTH, this parameter is not allowed.

See also the KEEPBACKV parameter.

Syntax: BAKDAY=*nnnn*

The variable *nnnn* is a number in the range 0-9999.

Default: None

CAL=

Purpose: Specifies whether aging is adjusted with the non-working days specification in the MAINVIEW SRM calendar definition. CAL=NO specifies that adjustments are not applied; CAL=YES specifies that date calculations take into account all non-working days in the MAINVIEW SRM calendar. This is an optional parameter.

Syntax: CAL=*YES/NO*

Default: CAL=NO

CDS=

Purpose: Specifies the type of DFHSM control data set to be affected. This is a required parameter.

Syntax: CDS=*xxxx*

The variable *xxxx* is a value as follows:

BCDS	backup control data set
MCDS	migration control data set
BOTH	backup and migration control data sets

Default: None

CREDDAYS=

Purpose: Specifies the age (number of days since creation of the original pre-migration data set) at which the migrated data set (and MCDS entry) is deleted. If CDS=MCDS or CDS=BOTH, then one and only one of these three parameters must be specified: CREDDAYS, MIGDAYS, REFDAYS. If CDS=BCDS, this parameter is not allowed.

Syntax: CREDDAYS=*nnnn*

The variable *nnnn* is a number in the range 0-9999.

Default: None

DSN=

Purpose: Specifies a data set name or name mask. All MAINVIEW SRM name masking parameters can be used. This is a required parameter.

Syntax: DSN=*xxxxxxxxxxxxxxxxxxxx . . .*

The variable *xxxxxxxx* specifies a data set name up to 44 characters long.

Default: None

EVENTID=

Purpose: Specifies the identifier assigned to a user event in an SMEVNT $_{xx}$ member. This parameter causes an event to be generated from this function.

Note: If EVENTID= is used on an FLST SET statement with MODE=INACT, the event is still issued.

Syntax: EVENTID= $xxxxx$

The variable $xxxxx$ is the 5-character string specified on the EVNTID parameter in the SMEVNT $_{xx}$ member.

Default: None

KEEPBACKV=

Purpose: Specifies the number of backup versions of a data set that should be kept. If not specified, all backup copies are deleted. Use this parameter only if CDS=BCDS.

Syntax: KEEPBACKV= nnn

The variable nnn is a number in the range 0-255.

Default: None

MIGDAYS=

Purpose: Specifies the age (number of days since migration of the data set) at which the migrated data set (and MCDS entry) is deleted. If CDS=MCDS or CDS=BOTH, then one and only one of these three parameters must be specified: CREDDAYS, MIGDAYS, REFDAYS. If CDS=BCDS, this parameter is not allowed.

Syntax: MIGDAYS= $nnnn$

The variable $nnnn$ is a number in the range 0-9999.

Default: None

POOL=

Purpose: Specifies the name of a single MAINVIEW SRM pool. Name masking cannot be used. If you specify the POOL parameter, only data sets from the specified pool are selected. This is an optional parameter. If you do not specify the POOL parameter, data sets are selected for processing without regard to pool membership.

Syntax: POOL=*xxxxxxxx*

The variable *xxxxxxxx* is a pool name from the SMPOOL*xx* member.

Default: None

REFDAYS=

Purpose: Specifies the age (number of days since the original pre-migration data set was last referenced) at which the migrated data set (and MCDS entry) is deleted. If CDS=MCDS or CDS=BOTH, then one and only one of these three parameters must be specified: CREDDAYS, MIGDDAYS, REFDAYS. If CDS=BCDS, this parameter is not allowed.

Syntax: REFDAYS=*nnnn*

The variable *nnnn* is a number in the range 0-9999.

Default: None

Usage Notes

SMHSMUTL generates DFHSM control cards to delete entries (and associated data sets) from the DFHSM migration and backup control data sets. These control cards are formatted for execution by TSO.

For the migration control data set, the general format of the control card is

```
HSEND DELETE dsn
```

For the backup control data set, the general format of the control card is

```
HSEND BDELETE dsn
```

SMHSMUTL processing is controlled by DELETE requests. Each delete request specifies the following information:

- data set name or name mask
- type of DFHSM control data set to be affected (migration or backup)
- aging specification (number of days after which the data set is to be deleted)

These criteria are compared to DFHSM control data set entries; eligible data sets cause the generation of a DFHSM delete control card. For example, the following SMHSMUTL control card

```
DELETE CDS=MCDS DSN=TD*.SPFTEMP*./ MIGDAYS=10
```

might generate the following control cards:

```
HSEND DELETE TDMOORE.SPFTEMP2.CNTL
HSEND DELETE TDCAREY.SPFTEMP1.LIST
HSEND DELETE TDJOHNSN.SPFTEMP0.CNTL
HSEND DELETE TDJOHNSN.SPFTEMP1.CNTL
HSEND DELETE TDJOHNSN.SPFTEMP2.CNTL
```

Up to 3,140 DELETE requests can be submitted in a single run.

MAINVIEW SRM must be active for SMHSMUTL to run. The generated control cards must be submitted to TSO to actually change the DFHSM control data sets.

Note that the DFHSM DELETE command deletes a migrated data set without recalling it. Both the MCDS entry and the migrated data set are deleted. The DFHSM DELETE command does not delete backup versions of the data set. The DFHSM BDELETE command deletes all or specific backup versions of a data set.

Control Card Syntax

The following is a list of guidelines for specific control card parameters:

- The DELETE keyword begins a specification. All parameters preceding the next DELETE keyword or the end of the input file are part of the specification.
- Parameters can be on the same line or separate lines. There is no line continuation character.
- Parameters can start in any column.

- Comments must be on separate lines. A comment must have an asterisk in column 1.
- Blank lines are allowed.

Name Masking

All MAINVIEW SRM name masks can be used with the DSN parameter. These mask characters are fully described in the *MAINVIEW SRM User Guide and Reference*. Briefly, the mask characters are as follows:

%	Any numeric character
?	Any character except period (.)
/	Any number of characters, to the end of a name
*	Any number of characters in a single name qualifier
**	Any non-zero number of name qualifiers

SMHSMUTL does not support the use of MAINVIEW SRM variables.

Messages

SMHSMUTL issues the following messages to SYSOUT (SMHSMPRM is the parameter processing program in SMHSMUTL):

```
SMHSMPRM REQUIRED PARAMETER MISSING: parameter
SMHSMPRM INVALID PARAMETER: parameter
SMHSMPRM DUPLICATE PARAMETER: parameter
SMHSMPRM INVALID VALUE FOR PARAMETER: parameter
SMHSMPRM CDS=MCDS/BOTH NOT ALLOWED WITH BAKDAYS
SMHSMPRM CDS=BCDS ONLY VALID WITH BAKDAYS
SMHSMPRM MAXIMUM OF 3140 DELETE STATEMENTS REACHED
```

Job Required for the SMHSMUTL Utility

In the JCL following, STEP10 executes SMHSMUTL to generate DFHSM control cards; STEP20 executes TSO in batch to have DFHSM process the cards. The DD names are defined as follows:

- STEPLIB specifies the library that contains MAINVIEW SRM load modules
- BCDS specifies the DFHSM backup control data set
- MCDS specifies the DFHSM migration control data set
- HSMCMD specifies an output data set for the generated control cards
- SYSIN specifies the input data set containing control cards for SMHSMUTL

```

//STEP10      EXEC  PGM=SMHSMUTL,REGION=2M
//STEPLIB      DD   DSN=EMP.?prefix.BBLINK,DISP=SHR
//SYSPRINT     DD   SYSOUT=*
//BCDS         DD   DSN=dfhsm.backup.control,DISP=SHR
//MCDS         DD   DSN=dfhsm.migration.control,DISP=SHR
//HSMCMD       DD   DSN=&&HSMOUT,DISP=(NEW,PASS),
//              SPACE=(TRK,(10,10)),UNIT=SYSDA,
//              DCB=(LRECL=80,BLKSIZE=13680,RECFM=FB)
//SYSIN        DD   *
DELETE DSN=**.*.TRAN* CDS=MCDS MIGDAYS=20
/*
//STEP20      EXEC  PGM=IKJEFT01
//SYSTSPRT     DD   SYSOUT=*
//SYSTSIN      DD   DSN=&&HSMOUT,DISP=(OLD,DELETE)

```

The following example shows syntax for the SMHSMUTL batch program.

DELETE CDS=MCDS DSN=**.*.SRCHFOR./ REFDAYS=10	Delete migration entries for search-results data sets after 10 days of non-use.
DELETE CDS=MCDS DSN=AP*.*.TRANS* CREDDAYS=90	Delete migration entries for accounts-payable-transaction data sets 90 days after creation.
DELETE CDS=BOTH DSN=/ MIGDAYS=180	Delete all backup and migration entries 180 days after migration.
DELETE CDS=BCDS DSN=GLYTD*.*.UPD* BAKDDAYS=20 KEEPBACKV=5	Delete all but 5 backup versions of genera- ledger-update data sets 20 days after backup.

Chapter 3 Space Collector

This chapter presents the following topics:

Overview	3-2
How the Space Collector Works	3-2
Using the Copy/Merge Utility	3-3
Controlling the Space Collector	3-6
Activating and Deactivating the Space Collector	3-8
Using More Than One Space Collector	3-10
Implementing Security	3-11

Overview

MAINVIEW SRM space utilization views assist you in determining the current use and growth of DASD in your data center. DASD usage can be reported from several different views, including overall summary by time, by storage pools, by RAID volumes, by applications, and by volumes.

How the Space Collector Works

Space utilization data is collected by the space collector and stored in the space database. The space collector writes a snapshot to a linear data set that maintains storage utilization information by volume, pool, and application. The period for which information is available is determined by the frequency of the snapshots as well as the size of the space database. Application-level information can be extracted from the application database.

The space database consists of three linear data sets: volume, pool, and application. Each data set contains a series of snapshots of DASD utilization taken from different perspectives. A fourth data set also is allocated. This fourth data set is not used at present but is required to ensure upward compatibility with future releases.

The *volume* data set contains a series of volume snapshots. The volume snapshots are created at regular intervals from the information that is available in the volume table of contents (VTOC) on each DASD volume.

The *pool* data set contains a series of pool snapshots. A pool is a user-defined group of DASD volumes that are monitored as a single entity. Volume pool definitions are defined using the data collector input parameters.

The *application* data set contains a series of snapshots that are extracted from the application database. The application collector provides real-time monitoring, budgeting, and control of DASD space utilization. Applications are user-defined and can be used to track space utilization by developers, project teams, applications, and departments. For more information about the application collector, see Chapter 5, “Application Collector.”

The space collector writes to the database using a wraparound method. When a data set is full, the earliest snapshots are overwritten by the current snapshot.

Using the Copy/Merge Utility

The space collector provides the Copy/Merge utility SGDCOPY, which enables you to perform the following tasks:

- use historical space databases from previous versions of the product
- expand the historical space database to accommodate new functionality
- create a backup for the active historical space database
- reorganize the database if the historical space data collector encounters a name table overflow

You can tailor the Copy/Merge utility so that the time interval and the time step between snapshots become a subset of the input snapshots. This utility is particularly useful when you are creating a backup for the active historical space database (for example, you are requesting a calendar month or year information).

You must use the Copy/Merge utility to reorganize the database if the historical space data collector encounters a name table overflow (message SGRD64E). Sample JCL is in the SGDCOPYJ member in *?prefix.BBSAMP*. Copy SGDCOPYJ to UBBSAMP before you modify it for your site.

You can also use the Copy/Merge utility to expand the historical space database to accommodate new functionality if you are upgrading from a release of the product before version 3.1.

Note: Expanded databases, after being updated with RAID and/or RVA information, cannot be used by previous releases of the product.

Warning! Do not use SGRDCOPYJ on empty or unused files, which can result in looping.

Copy/Merge Processing Options

Processing options tell the Copy/Merge utility the kind of processing that it should perform. You must specify the processing options through the PARM= job control parameter.

You define each processing option by using a keyword parameter. The following keywords, listed in alphabetical order, are available for Copy/Merge processing.

Note: Specify each parameter only once.

ACTION=COPY | BACKUP

This optional parameter defines the action that the Copy/Merge utility must take when the first defined extent fills during the writing of the output data set.

ACTION=COPY specifies that the Copy/Merge utility should behave just like the data collector. That is, the Copy/Merge utility should perform a wraparound when the first extent is full on the output data set, thus overwriting the oldest snapshot with the next one. This action ensures that the defined size for the output data set is used. Refer to the *MAINVIEW SRM Customization Guide* for a worksheet for calculating the database size.

ACTION=BACKUP is used when no wraparound should occur. In other words, the Copy/Merge utility allows secondary extents to be allocated when the first extent fills. This option is used when you do not want to calculate the exact size of the data set containing a given amount of information or when you do not want to lose space by allocating more size than absolutely necessary.

The default value is **COPY**.

Abbreviations: **Action=**

BEGIN=*date of oldest snapshot to be copied*

This optional parameter defines the date of the oldest snapshot that should be included in the output data set. Snapshots that were created earlier than the specified date are excluded from the copy operation.

The following formats are supported:

YY.DDD
YYYY.DDD
DD/MM/YY
DD-MM-YY
DD.MM.YY
DD/MM/YYYY
DD-MM-YYYY
DD.MM.YYYY

If this parameter is omitted, no filtering for the oldest record takes place.

Abbreviations: **BEGin=**

END=*date of last snapshot to be copied*

This optional parameter defines the date of the latest (most recent) snapshot that should be included in the output data set. Snapshots that were created after the specified date are excluded from the copy operation.

The following formats are supported:

YY.DDD
YYYY.DDD
DD/MM/YY
DD-MM-YY
DD.MM.YY
DD/MM/YYYY
DD-MM-YYYY
DD.MM.YYYY

If this parameter is omitted, no filtering for the most recent record takes place.

MAXACCNTCODES=*number of account codes used*

This optional parameter defines the number of different account codes that the output data set must accommodate. The value you enter must be between 325 and 65535.

The default value is 325.

Abbreviations: **MAXACCNTCODEs**=, **MAXACCTCODEs**=

MININTERVAL=*minimum time interval between permanent snapshots in minutes*

This optional parameter is used to determine the minimum time step in minutes between snapshots that are written to the output data set. This parameter is used only to exclude snapshots that are already present on the input, thus allowing the same data set size to cover a longer (although less detailed) history. In particular, **MININTERVAL** has meaning only when it defines a greater value than the time interval that is used on the input data set (very likely determined through the **WRITEINTERVAL** parameter on the data collector).

If you define a value that is not greater than the value in effect in the input data set, or if you omit this parameter, all complete snapshots are copied to the output data set.

The value you enter must be between 1 and 1440.

Abbreviations: **MININTerval**=, **MINNTVI**=, **INTERVal**=, **NTVL**=

SMFID=0 or SMF record number

This parameter tells the Copy/Merge utility which SMF records to select when it reads SMF input. The same record number must be defined for this keyword as was defined earlier for the data collector. This parameter is required and is allowed only when reading SMF input.

Abbreviations: **SMFid=**

TYPE=VOLUME | POOL | ACCOUNT

This parameter tells the Copy/Merge utility the type of records to be copied. This parameter must always be defined.

VERSION=1.1 | 2.1 | 2.2 | 2.3 | 2.4 | 2.5 | 3.1 | 4.1 | 5.1 | 6.1 | 7.1 | 7.2

This optional parameter tells the Copy/Merge utility which version of space collector (formerly the StorageGUARD product) input records should be copied or converted. The output data set is always formatted according to the most recent version. The Copy/Merge utility verifies that the input records correspond to the version that is defined (or used by default). Record (format) conversion is performed as required when you define releases older than the current one.

The default value is the most recent version.

Abbreviations: **VERsion=**

Controlling the Space Collector

You use system parameters and pool definitions to control the space collector. Use the input parameters to specify the volumes to be monitored, how pools are constructed, and the read and write intervals.

The following system parameters are specified in the SMMSYS $_{xx}$ member of *?prefix.BBPARM*. The parameters are described in the *MAINVIEW SRM Reference Summary*.

Note: The space data collector collects information for volumes that have been assigned to a pool only. See the *MAINVIEW SRM User Guide and Reference* for instructions on how to define a pool.

Table 3-1 describes the system parameters that you can use to control the space collector.

Table 3-1 Space Collector System Parameters (Part 1 of 2)

Parameter	Description	Default
SG_INITPOOL=	specifies the maximum number of defined pools included in a single snapshot	1000
SG_INITVOL=	specifies the maximum number of defined volumes included in a single snapshot	3000
SG_MAXACCT=	specifies the maximum number of active accounts in the application database	extracted from application database
SG_READNTVL=	specifies the frequency (in minutes) at which the space collector scans the DASD volumes for historical space information to create a snapshot in memory	30
SG_RETRYLIM=	specifies the number of abend conditions that the space collector should ignore	10
SG_SPACHLDR=	defines a data set name mask that the space collector can use to identify space holder data sets	none
SG_SUBTASKS=	specifies the number of volumes that can be read in parallel	3
SG_WRITNTVL=	specifies the frequency (in minutes) at which snapshots are written to the space database	30
SGD_PROCNM=	specifies the name of the data collector started task (1-8 characters)	SGDCOLLS
SGD_SMFID=	specifies an SMF record number for MAINVIEW SRM audit records that are written to the SMF data set for the space collector	0
SGDCOLLECT=	indicates the default value for the pool collection; a corresponding parameter at the POOL level overrides the default	N
SGDCOLLECT n =	specifies whether a pool is processed by the space alternate data collector that is assigned a suffix of n , where n is a number in the range 1-8.	none
SGDPROCNM n =	specifies the cataloged procedure to be started for a specified copy of the space database	none
SGDSMFID n =	specifies an SMF record number for MAINVIEW SRM audit records that are written to the SMF data set for a specified copy of the space database	none
SGINITPOOL n =	specifies the maximum number of defined volumes to be included in a single snapshot for a specified copy of the space database	none
SGINITVOL n =	specifies the number of hours between refreshes of the IXFP data tables for a specified copy of the space database	none
SGMAXACCT n =	specifies the maximum number of active accounts in the application database for a specified copy of the space database	extracted from application database
SGMAXPOOL n =	specifies the number of pools that can be assigned to a volume for a specified copy of the space database	none
SGMAXSSDSZ n =	specifies the maximum number of cylinders used for a solid state disk drive for a specified copy of the space database	0
SGREADNTVL n =	specifies the frequency at which the space collector creates a snapshot for a specified copy of the space database	30

Table 3-1 Space Collector System Parameters (Part 2 of 2)

Parameter	Description	Default
SGRETRYLIM n =	specifies the number of abend conditions that the space collector should ignore for a specified copy of the space database	10
SGSPACHLDR n =	defines a data set name mask that the space collector can use to identify space-holder data sets for a specified copy of the space database	none
SGSUBTASKS n =	specifies the number of volumes that can be read in parallel for a specified copy of the space collector for a specified copy of the space database	3
SGWRITNTVL n =	defines the frequency at which snapshots are written to the space database for a specified copy of the space database	30
START_SGD=	can be set to N to prevent the automatic startup of the space collector	Y

Activating and Deactivating the Space Collector

The space collector is activated when the Reporting component of MAINVIEW SRM is started with SVOS and is deactivated when the Reporting component is stopped by using SVOS.

Note: All data-collector load modules must be located in an APF-authorized load library.

The JCL that is used to activate the space database data sets is in *?prefix.BBSAMP* (SGDCOLLS). The started task name is specified by using the SGD_PROCNM parameter in the SMMSYS xx system member. Modify SGDCOLLS according to the instructions in the member header, and then copy the member to a system procedure library.

Tip: Ignore any IEC999I IFGOTC0A, IFGOTC0B messages that occur during shutdown. These informational messages may appear when the data collector terminates an RVA collection subtask.

Table 3-2 presents the requirements for the data definition (DD) statements that are used in SGDCOLLS.

Table 3-2 Requirements for SDCOLLS DD Statements (Part 1 of 2)

Data Definition	Contents	Requirements
SGRDVOL	volume snapshots	This DD statement is required and the data set should be allocated with a DISP=SHR.
SGRDPOOL	pool snapshots	This DD statement is required and the data set should be allocated with a DISP=SHR.

Table 3-2 Requirements for SCDCOLLS DD Statements (Part 2 of 2)

Data Definition	Contents	Requirements
SGRDACNT	application snapshots	This DD statement is required and the data set should be allocated with a DISP=SHR.
SGRDDSN	n/a	Reserved for future use. This DD statement is required and the data set should be allocated with a DISP=SHR.
SGCDB	application collector application database	This DD statement is required and the data set should be allocated with a DISP=SHR. If you do not use the application collector, you can code //SGCDB DD DUMMY.
SIBLMSG	IXFP SIBBATCH output messages work file	This DD statement is required only when you are collecting IXFP RVA device information.
SIBRMSG	IXFP SIBBATCH output reports work file	This DD statement is required only when you are collecting IXFP RVA device information.
STEPLIB	space collector load modules	Defines the <i>?prefix</i> .BBLINK library that contains the space collector load modules, and the user-defined <i>?prefix</i> .USER.BBLINK library. User exits should be placed in an authorized, user-defined <i>?prefix</i> .USER.BBLINK library. When you are collecting IXFP RVA device information the following IXFP load libraries must be included in the link list or added to the STEPLIB concatenation: ?ixfphq.SIBLOAD ?ixfphq.STKLOAD ?ixfphq.SIBLINK
SVWEXEC	space collector REXX procedures	Defines the <i>?prefix</i> .BBCLIB library that contains the REXX procedures. This DD statement is required only when you are collecting IXFP RVA device information.
SYSIN	IXFP SIBBATCH control statements work file	This DD statement is required only when you are collecting IXFP RVA device information.
SYSPRINT	IXFP SIBBATCH system messages output file	This DD statement is required only when you are collecting IXFP RVA device information.
SYSTEM	IXFP SIBBATCH terminal output file	This DD statement is required only when you are collecting IXFP RVA device information.
SYSTSPRT	system output file	This DD statement is required when you are collecting IXFP RVA device information. This system output data set is used by the REXX interpreter for error messages, as well as output from the REXX SAY command.

Using More Than One Space Collector

In most cases, a single data collector is sufficient for all of your installation's purposes. If you find a significant need to run one or more additional data collectors, use the sample SGDCOLLX in *?prefix.USER.BBLINK*. Place a copy of this new procedure in a standard accessible PROCLIB data set and modify it as described for SGDCOLLS (see "Activating and Deactivating the Space Collector" on page 3-8). The JCL in SGDCOLLX differs from the JCL in SGDCOLLS because SGDCOLLX contains a symbolic parameter that is used in the PARM field for the SGRDCOLL program.

Assign a unique number (1–8) to each alternate collector. Then, specify the name of the procedure as the value on the SGDPROCNM n parameter.

Example

If you assigned the number 3 to an alternate collector, the parameter and value for it would be SGDPROCNM3=procedure name. You then define other parameters (for example, SGMAXTASKS3, SGREADNTVL3, and SGMAXPOOL3) to specify any differences from the default values.

You must also use SGDCOLLECT3=YES in the system pool member to identify the pools that should be monitored by this collector. Only pools that are explicitly designated are monitored by an alternate collector. (See the *MAINVIEW SRM Reference Summary* for descriptions of these parameters.) The symbolic SVSGD3 would be used in an SVOS start command to invoke the alternate collector.

Each data collector must have its own copy of the data collector data sets. Data collector data sets cannot be shared between collectors. To control which collector supplies data to views, specify the collector number in the SETSRM keyword SGDID (see Appendix A, "SETSRM Command" for more information).

The following values are set by SVOS when the Reporting component is started in SVOS:

SSID=ssss

ssss is the subsystem ID assigned to SVOS in the STC JCL PARM field,

PARM='SUBSYS=ssss'

SGDID=*n*

n is the alternate collector ID number

This number is assigned by SVOS when the product is started. The SVSGD default is 0 in the JCL; default values for SVSGD1–8 are 1–8. For more information, see the description of the SGDPROCNM*n*= parameter in the *MAINVIEW SRM User Guide and Reference*.

Implementing Security

The data collector must have sufficient authorization to be able to collect information about all of volumes in the system. However, the data sets that are involved in data collection must be secured against unauthorized modifications.

The data collector must be able to access all data sets that are assigned to it. In addition, it requires update access to its own databases (DD names SGRDVOL, SGRDPOOL, SGRDACNT and SGRDDSN).

All users who are authorized to use MAINVIEW SRM should have read access to the data collector data sets. Storage management staff may also require read access to the data sets that are used by the REXX procedures. These data sets contain a log that describes exceptional conditions that were encountered, information about the jobs that were submitted to background, and so on.

It is important to prevent simultaneous updates of data sets. You cannot use the standard method that is provided by OS/390 to give the data collector exclusive control of the data sets. Under OS/390, no other task could read these data sets while the task (which exclusively owns the data sets) is running. But you may want to read both the database and the other data sets for current information. Since the data collector should always be running whenever your system is up, one method of dealing with this problem involves allocation of data sets as DISP=SHR. This removes the protection that OS/390 would otherwise provide.

Note: You must make sure that no improper modification occurs. Update access to the database, logs, and so on should *not* be assigned to users of the data.

Use caution when you run multiple data collectors. Each data collector must have a dedicated data set of its own. No standard technology prevents you from starting the same task several times in your processor complex, but a simple method provides help. You can assign a separate data set to each data collector with exclusive control. These data sets can be allocated with no space occupied at all (with SPACE=(TRK,0)) because they serve only to activate the OS/390 control mechanism that ensures data set integrity.

Use this mechanism to make sure that no other task can allocate this data set when the data collector is active. This technique delays the second (accidental) start of a procedure that is already running on a processor. This simple method cannot be used on different processors unless you have Global Resource Serialization (GRS) or a similar product that expands the same protection mechanism to cover the whole enterprise. It is the responsibility of operating personnel to make sure that the same procedure is never active on more than one processor.

Chapter 4 Performance Collector

This chapter presents the following topics:

Overview.....	4-2
How the Performance Collector Works	4-2
Activating SMF and CMF/RMF Parameters	4-2
Controlling the Performance Collector.....	4-5
Filtering Performance Data.....	4-6

Overview

Performance data assists you in determining the impact that current use and trends in storage use have on the performance of the storage resources in your data center. Both the absolute utilization of physical volumes and the allocation of specific data sets can cause response time changes. Being able to pinpoint problems when they occur, as well as project when problems are likely to occur, provide storage administrators with information for managing their DASD resources. This is even more important for the optimization of high-capacity, high-performance RAID implementations. MAINVIEW SRM performance provides access to all saved performance data that the performance collector gathers.

How the Performance Collector Works

By default, the performance collector is started at system startup. You can control the startup of the collector with the `START_SGP` system parameter, or you can deactivate and activate the collector using the `COMPSTAT` view.

The performance database is the repository that is used to store data that the performance collector gathers. You can choose to store data in a maximum of 100 databases before old data is overwritten. There is never more than one *active* performance database, but at system startup, previously collected data from all databases can be accessed.

The information collected by the performance collector is LPAR-specific. Therefore, the performance collector must run as a started task on each LPAR, and each LPAR must have a unique performance database.

Activating SMF and CMF/RMF Parameters

To collect performance data, the following requirements must be met:

- SMF and CMF or RMF must be active.
- SMF must have the IEUF83 and IEUF84 exits specified for the areas for which data collection is desired.
- The SMF global recording interval should be specified unless the system default of 30 minutes is acceptable. The SMF global recording interval should be synchronized with some part of the hour unless the system default of 00 is acceptable.

- The CMF or RMF recording interval duration must be equal to or an even multiple of the SMF recording interval duration. For example, if the SMF recording interval duration is 30 minutes, the CMF or RMF recording interval may be 5, 10, 15, or 30 minutes.
- The CMF or RMF recording interval must be synchronized to the same part of the hour as SMF.
- SMF types 30, 42, 73, 74, and 78 records must be turned on. The SGP_SMF42 system parameter in the SMMSYS.xx member can control the actual recording of type 42 records in the SMF data set.

Warning! Due to limitations in RMF and CMF, BMC Software cannot gather I/O queuing information statistics on a VM guest.

CMF Requirements for Data Collection

The recording interval of either CMF or RMF must be equal to or an even multiple of the SMF recording interval. The following parameters are the minimum required for CMF:

```
RECORD INTERVAL=xx , RUNTIME=1440 , SMF=YES , SYNCH=yy
CHANNEL
DEVICE CLASS=DASD
```

In this example, *xx* is equal to or an even multiple of the SMF recording interval duration; *yy* is equal to the SMF recording interval part of the hour synchronization value.

Table 4-1 lists the keywords required in the CMFCPMxx member to collect the specified records.

Table 4-1 Required Keywords in CMFCPMxx

Record Type	Required Keyword	Data Collected
Type 73	CHANNEL	channel activity
Type 74-1	DEVICE	device activity
Type 74-2	XCFDATA	cross-system coupling facility (XCF)
Type 74-3	OMVS	open edition MVS
Type 74-4	CFDATA	coupling facility
Type 74-5	CACHE	cache data records

Table 4-1 Required Keywords in CMFCPMxx

Record Type	Required Keyword	Data Collected
Type 78-1	IOQ	I/O queuing data for 4381 and 3080 processors
Type 78-2	VSMDATA	virtual storage data

RMF Requirements for Data Collection

If you are running RMF, the parmlib member must specify the following measurement, timing, and recording options:

Measurement: CHAN
 DEVICE(DASD)
 Timing: SYNC(SMF) (synchronizes SMF/RMF intervals)
 Recording: RECORD

To create RMS post processor-compatible 74-5 records, you must stipulate the following in CMFCPMxx:

```
CACHE CMFREC=74
```

CACHE CMFREC=74 generates 74-5 SMF records and 240-27 records, whereas, the CACHE keyword used alone generates 240-27 records for the BMC Software proprietary cache reports.

SMF Requirements for Data Collection

The following example shows how to set up SMF for the proper exits and recording of SMF records.

```

INTVAL(15)                /*GLOBAL INTERVAL)*/
SYNCVAL(15)              /*Synchronization value*/
ACTIVE                   /*ACTIVE SMF RECORDING*/
DSNAME(SYS1.MAN1,SYS1.MAN2,SYS1.MAN3) /* THREE DATA SETS */
NOPROMPT                 /*DO NOT PROMPT OPERATOR FOR OPTIONS*
REC(PERM)                /*TYPE 17 PERM RECORDS ONLY*/
MAXDORM(3000)           /* WRITE AN IDLE BUFFER AFTER 30 MIN*
STATUS(010000)         /* WRITE SMF STATS AFTER 1 HOUR*/
JWT(0800)               /* 522 AFTER 8 HOURS */
SID(SYSG)               /* SYSTEM ID IS SYSG */
LISTDSN                 /* LIST DATA SET STATUS AT IPL*/
LASTDS(MSG)            /*DEFAULT TO MESSAGE */
NOBUFFS(MSG)           /*DEFAULT TO MESSAGE */
SYS(EXITS(IEFU83,IEFU84,IEFACTRT,IEFUJV,
          IEFUSI,IEFUJI,IEFUTL,IEFU29),INTERVAL(SMF,SYNC),
      NODETAIL)
SUBSYS(STC,EXITS(IEFU29,IEFU83,IEFU84,IEFUJP,IEFUSO),
      INTERVAL(SMF,SYNC))
SUBSYS(XXXX,EXITS(IEFU29,IEFU83,IEFU84,IEFUJP,IEFUSO),
      INTERVAL(SMF,SYNC)) /*XXXX IS SUBSYSTEM CMF TASK IS USING */

```

Controlling the Performance Collector

Table 4-2 describes the system parameters that control the performance collector.

Table 4-2 Performance Collector System Parameters (Part 1 of 2)

Parameter	Description	Default
SGP_MAXDSNS	maximum number of data sets that could be referenced during a recording interval duration	1000
SGP_MAXVOLS	maximum number of volumes that could be referenced during a recording interval duration	250
SGP_MAXJOBS	maximum number of jobs that could be referenced during a recording interval duration	200
SGP_MAXSCLS	maximum number of storage classes that could be referenced during a recording interval duration	100
SGP_MAXPOLs	maximum number of pools that could be referenced during a recording interval duration	100
SGP_MAXPTHS	maximum number of channel paths that could be referenced during a recording interval duration	256

Table 4-2 Performance Collector System Parameters (Part 2 of 2)

Parameter	Description	Default
SGP_MAXLCUS2	maximum number of logical control units that could be referenced during a recording interval duration If the value is set too low, the system does not set aside enough buffer space to handle all of the records. You must determine a value that provides enough buffer space for LCU records without causing a shortage of dataspace storage for other records.	256
SGP_MAXCCUS	maximum number of cache control units that could be referenced during a recording interval duration	256
SGP_MAXDIRS	maximum number of RAID EMC directors that could be referenced during a recording interval duration	256
SGP_MAXPVLS	maximum number of RAID EMC physical volumes that could be referenced during a recording interval duration	250
START_SGP	can be set to N to prevent the automatic startup of the performance collector	Y

Filtering Performance Data

Collection of performance data uses the MAINVIEW SRM filter list capability to identify the data that should be recorded in the performance database. Each function is defined in *?prefix*.BBPARAM member SMFUNCxx. Each definition, except SGPFILTR, controls the inclusion or exclusion of a particular record type. Each function is defined as ACTIVE=NO in SMFUNCxx. If a function is left inactive, no records of that type are recorded in the performance database.

Each function is defined in a filter list SMFLSTxx member in *?prefix*.BBPARAM. Each filter list member specifies that all records of a particular type are to be recorded in the performance database when the function is activated.

In general, to use a function, perform the following steps:

- Step 1** Set the ACTIVE parameter to YES in the function definition in SMFUNCxx.
- Step 2** Define a filter list in an SMFLSTxx member with MODE=ACT and with parameters to apply the function to a set of resources (jobs, data sets, volumes, pools, and so on).

Chapter 5 Application Collector

This chapter presents the following topics:

How the Application Collector Works	5-2
Check Processing	5-2
Add Processing	5-4
Modes of Operation	5-4
Creating Applications	5-5
Dynamically Creating an Application	5-6
Manually Creating an Application	5-7
Maintaining Budgets	5-7
Assigning Applications to a Data Set	5-8
Reviewing and Updating Applications	5-8
Reporting	5-9
TSO BUDGET Command as a Batch TMP	5-10
DFHSM Migration and Backup Report	5-11
SAS Reporting Procedures	5-12
COBOL Reporting Programs	5-13
Reporting with the SGCCOB01 Member	5-13
Trend Analysis Reporting	5-17
TSO Commands	5-18
BUDDSN Command	5-18
BUDGET Command	5-21

How the Application Collector Works

The application collector is a subcomponent of MAINVIEW SRM Reporting that provides these benefits:

- minimizes space-related problems
- maximizes the use of DASD space
- assists in the automation of DASD storage management

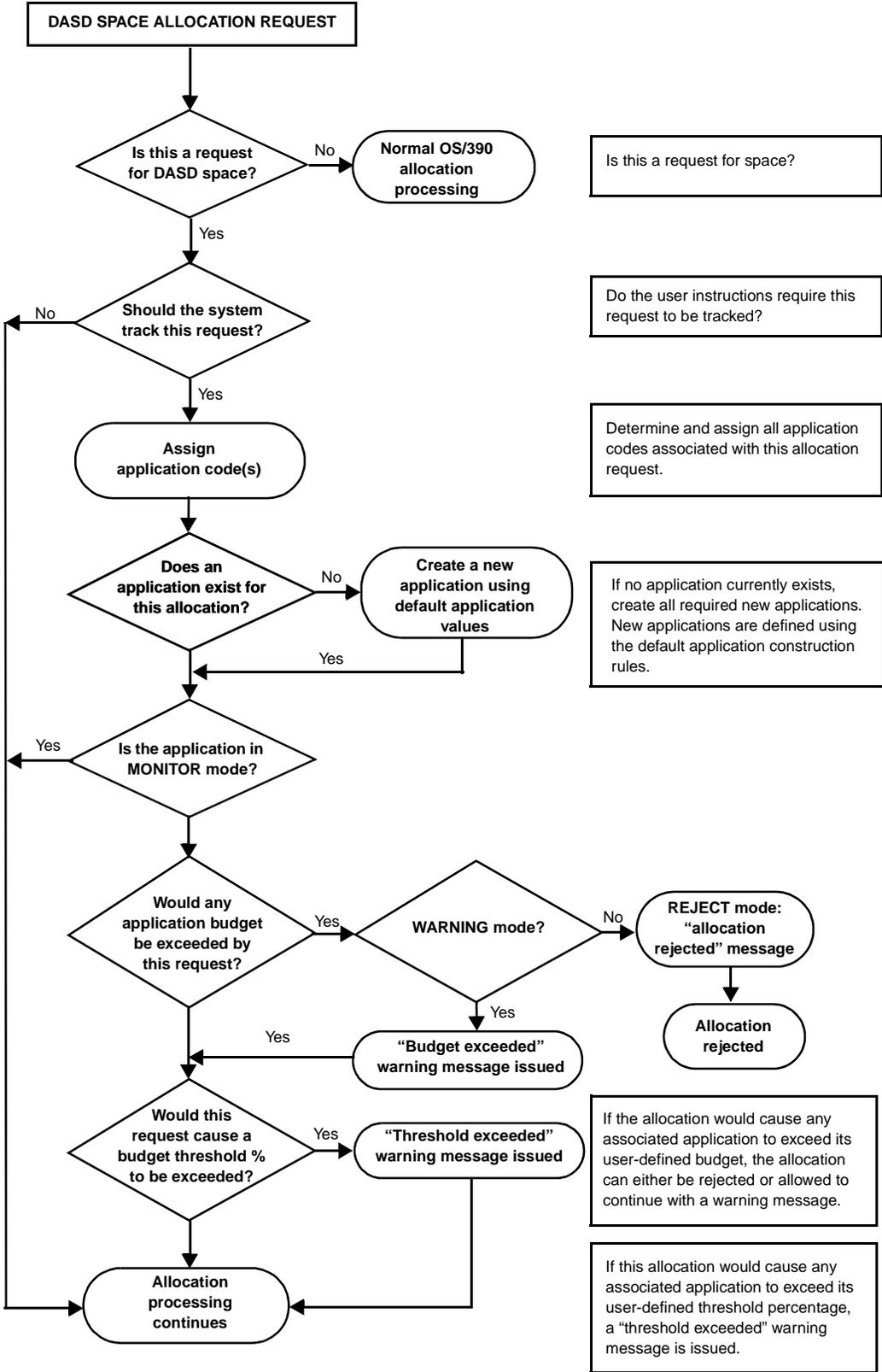
The application collector monitors space allocation and deallocation on DASD using the DADSM preprocessing (IGGPRES00) and postprocessing (IGGPOST0) exits. The collector works in conjunction with any existing DADSM exits your installation might have already established. The system uses the DADSM exits to monitor and control DASD space requests.

Figure 5-1 on page 5-3 and Figure 5-2 on page 5-4 illustrate how MAINVIEW SRM Reporting component uses the DADSM exits to monitor and control DASD space requests.

Check Processing

The DADSM preprocessing flowchart shown in Figure 5-1 on page 5-3 shows how MAINVIEW SRM Reporting component performs check processing. Check processing ensures that Reporting tracks only the desired space request, dynamically creates applications in the application database (if necessary), generates “over budget” and “warning threshold exceeded” messages, and optionally, rejects allocation requests.

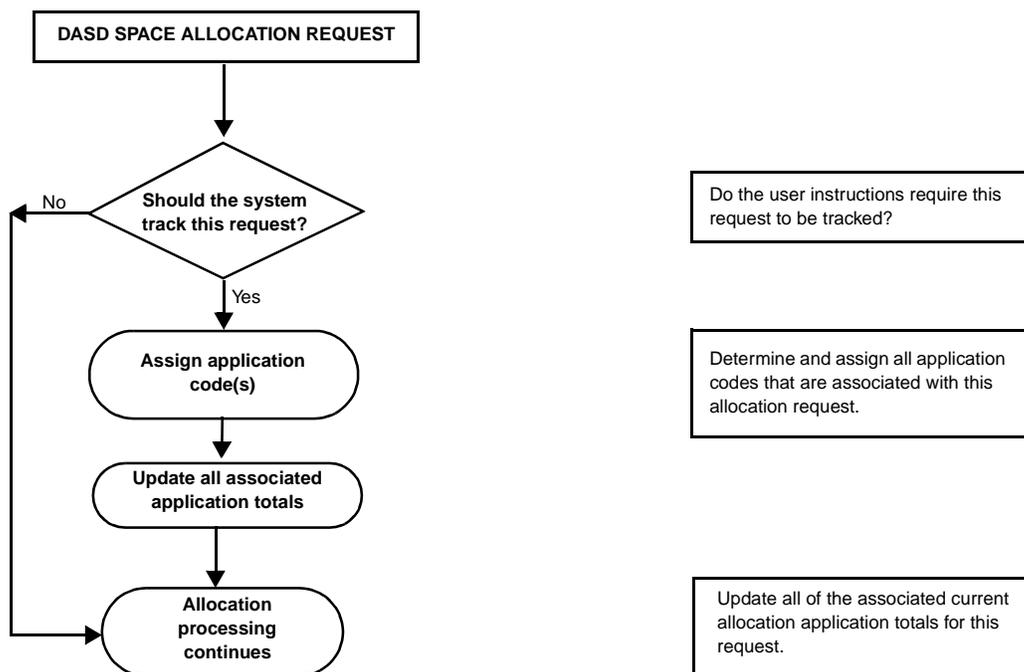
Figure 5-1 Check Processing



Add Processing

The DASD postprocessing flowchart shown in Figure 5-2 shows how MAINVIEW SRM Reporting component performs add processing. Add processing updates the current allocation totals for the associated applications.

Figure 5-2 Add Processing



Modes of Operation

The application collector can be run in a combination of modes: monitor, warning, or reject.

- **Monitor Mode**—BMC Software recommends that you initially implement your system in monitor mode and specify a budget for all applications running in monitor mode so that information on DASD space utilization is gathered. Budgets are used as the basis for percentage usage calculations and graphic and tabular displays. In monitor mode, the budget is used by MAINVIEW SRM Reporting component only for calculation reporting purposes. The information gathered in the reports helps to determine how space is being used in an installation and if there is a need for warning or reject mode.

- **Warning and Reject Modes**—Applications set to warning or reject mode may be assigned a budget for allocation. In warning mode, when the limits of the budget are approached or exceeded, a warning message is written to the SYSLOG and job log and sent to the TSO user (if the request was from a TSO session). In reject mode, a warning is also issued when the limits of the budget are approached. Additionally, if a space allocation request will cause an allocation total to exceed the budget, the request is rejected and reject messages are issued. See Table 5-1 for the effects of modes on application processing.

Table 5-1 Effects of Modes on Application Processing

System Action	Current Application Mode		
	MONITOR	WARNING	REJECT
Warning threshold percentage exceeded. Warning message sent?	N	Y	Y
“Over Budget” warning message is sent?	N	Y	N
Allocation rejected?	N	N	Y

Combining Modes

You can implement modes globally by specifying that they default as system options. The system option is stored in the application database. To override system options, set modes as application options when you set up an application. Specifying modes as application options allows variations in the space administration for different groups of users or types of work. It also allows for the phased implementation of warning and reject modes. Full implementation of modes requires the use of both system default and application modes.

Creating Applications

Use MAINVIEW SRM data entry panels to maintain applications and budgets. For more information about using data entry panels as well as how to use TSO commands and other facilities to manage applications, see the *MAINVIEW SRM Reporting User Guide*.

An application is a collection of information in the applications database that is related to an application code. This information includes the following data:

- **Current allocation totals**—amount of space currently allocated on DASD for this application code
- **High-water marks**—largest amount of space used or reserved for this application code since high-water marks were last reset
- **Budgets for permanent, temporary, and VSAM space**—maximum amount of space (as determined by your DASD space administrator) allowed for this budget application

A number of user fields are available that you can use to retain descriptive information and codes. For example, user fields can retain codes to collate information for reporting purposes.

You can create an application dynamically or use a TSO command to manually create an application.

Dynamically Creating an Application

You can create applications dynamically as needed. If the application collector constructs a new application code during the processing of a space allocation request, a new application is created automatically. The resynchronization function (RESYNC) also supports this feature. RESYNC is used to ensure that a database reflects DASD utilization accurately and provides a method of populating a new database with application information.

New applications created automatically can be customized using the TSO BUDGET command.

Manually Creating an Application

You can create applications manually by using the TSO BUDGET command. The values used to initialize the budgets, user fields, application options, and so forth, are obtained by one of the following methods:

- Values can be extracted from a default or model application. Typically, an installation establishes a single default application with an application code of CONTROL. Values are extracted from this application whenever a new application is created. Model applications can be created using the BUDGET TSO command. An active application also can be used as a model.
- Values can be provided by a user subroutine. The default exit allows an installation to determine programmatically the application code of the model application that should be used when creating an application automatically. The application code of the model application can be assigned based on the new application code data set name, data set organization, or volume serial. This subroutine also can be used to replace individual fields in the new application.
- Values can be specified using the TSO BUDGET command. The TSO BUDGET command also supports a FROM parameter that can be used to provide the application code of a model application. This facility allows the DASD space administrator to choose a model from a set of model applications when creating new applications manually.

Note: BMC Software recommends adopting a unique application-code naming convention to help identify model applications.

Maintaining Budgets

Use the BUDGET TSO command to change individual applications or to change groups of applications, as follows.

- Use a query statement to select applications that you want to change in groups. A query statement can be used to select applications based on the value in a field or by comparing two fields in the application record. For example, a BUDGET TSO command could be coded to change the mode and permanent allocation budget of all applications to 500KB where user field 1 contains the literal TEST ACCT.

- Individual applications are selected by using their application codes. Individual applications can be changed interactively, or a number of applications can be changed quickly and efficiently by running the budget commands under a batch TMP.

Assigning Applications to a Data Set

Numerous factors determine the applications associated with a data set. A data set can be associated with one to four application codes, or it can be ignored by the application collector. The following facilities can be used to determine the application code:

- **Default Application Code Construction** can concatenate components extracted from the data set name, data set organization, or volume serial to form an application code. The default application code construction rules are specified when an application database is created. These default rules apply to all space allocation requests. Using the high-level qualifier of the data set name is a common use of this feature.
- **Filter and Rule List Processing** supports the use of conditional logic to construct the application code and to determine if a space allocation request should be tracked by the application collector. This facility is especially useful when there is a need to group data sets with unlike or nonstandard names. Revised filter and rule lists can be activated at any time using the operator interface or an operator command.
- **User-Coded Subroutines** determine whether a space allocation request should be tracked and, if so, the associated application code.

Note: Though the application code build subroutine is rarely used, MAINVIEW SRM Reporting component includes support for user subroutines for the sake of completeness.

Reviewing and Updating Applications

You can review and update applications in real time by using the BUDGET TSO command. With the BUDGET TSO command, access to the applications is controlled by a user-written subroutine. This feature is used to centralize or distribute the management of the applications.

To access an individual applications, specify an application code or use the query feature to specify sets of applications. The query feature has the following capabilities:

- A query can select all applications with a description starting with PROD (for example) or all of the applications where the current allocation total for permanent space exceeds the budget amount.
- Options are provided to limit the amount of information returned for each application. A query can be specified to return only the application codes of applications over budget.
- Totals can also be calculated for all of the applications that are selected by a query. A query can be specified to calculate and display the total space allocated for all applications with a description starting with PROD.

Reporting

MAINVIEW SRM views provide online reporting for an individual application code. This feature offers you an up-to-the-minute picture of your DASD space utilization and availability.

The MAINVIEW SRM Reporting component also includes a number of sample COBOL report programs and SAS procedures. Since the reporting requirements vary by each installation, these programs and procedures are provided as models. Sample reports include the following information:

- alphabetic listing of applications
- largest user of DASD space
- applications over warning level
- summary of high-water marks
- application space utilization graph

You can use a series of trend report programs and SAS procedures to monitor the changes in DASD space utilization over a period of time.

The application database contains information that may be useful to various departments in your organization. The system includes monitoring programs that can use either the active or other versions of the database to produce reports. The monitoring facilities can be used to examine the following information:

- compare current allocation amounts to budget
- analyze current allocation amount in contrast to available space
- DASD billing based on current allocation amount and high-water marks
- DFHSM backup and migrate data set space utilization
- capacity planning using current allocation amount and/or high-water marks

In addition, information can be extracted from the application database at regular intervals to produce a trend analysis report.

You can use the following methods for monitoring the budget applications on your system and producing reports:

- Execute COBOL reporting programs. Sample programs are provided in the *?prefix.BBSAMP* library.

Tip: When using sample programs in the *?prefix.BBSAMP* library, you may want to copy the sample member to a user library for modification.

- Execute the TSO BUDGET command as a batch TMP.
- Execute SAS procedures. Sample procedures are included in the *?prefix.BBSAMP* library.
- Execute the DFHSM Migration and Backup Report batch program.

Each of these batch reporting tools is explained in detail in the remaining pages of this chapter.

TSO BUDGET Command as a Batch TMP

The TSO BUDGET command can display an individual application or a range of applications. Space totals can be accumulated for all of the requested applications. See “TSO Commands” on page 5-18 for more information on TSO commands.

For example, if all of the applications for the accounting department have a user field set to ACCTING, the BUDGET command can list all these applications and display the space totals. Also, you can suppress the individual applications and print only the totals.

To produce a batch report with the TSO BUDGET command, first develop the command under TSO. Then you can execute the BUDGET command in batch using IKJEFT01. The following example produces an ACCTING space report that displays only the space totals for the accounting department.

Note: This example assumes that the first user field has been set to ACCTING for all of the applications used by the accounting department. If you want to use other than the active application database, you can add a SGCDB DD statement.

Figure 5-3 BUDGET TSO Command Batch TMP

```
//JOBNAME JOB . . .
//IKJACCT EXEC PGM=IKJEFT01,DYNAMNBR=10
//STEPLIB DD DISP=SHR, DSN=<APFLIB>
//SYSPRINT DD SYSOUT=*
//SYSTEM DD SYSOUT=*
//SYSIN DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
PROFILE NOPREFIX
BUDGET (USERF1,EQ,ACCTING) LIST(NONE) TOTAL
/*
```

Figure 5-4 Sample Output

```
SGC3202 **TOTAL**
SGC3203 DATE-CREATED=2000/02/27 LAST-USED=2000/03/05 WARN= 0% INDEX=235
SGC3205 TYPE=PERM CUR= 5533.59K MAX= 1234.00K HWM= 22813.46K
SGC3205 TYPE=TEMP CUR= 0.00K MAX= 0.00K HWM= 0.00K
SGC3205 TYPE=VSAM CUR= 0.00K MAX= 0.00K HWM= 0.00K
SGC3205 TYPE=HSM CUR= 0.00K HWM= 0.00K
```

DFHSM Migration and Backup Report

The DFHSM Migration and Backup report enables installations that use DFHSM to monitor the usage of DFHSM space by each application. Like the DFHSM accounting in the system, the use of this report assumes that the application name is constructed from the first two qualifiers of the original data set name. The figures represented on the report are each application name.

Member SGCHSMJ1 of the *?prefix.BBSAMP* library contains sample execution JCL for this report.

The following sample report shows the number of kilobytes used by the migrated and backup data sets associated with each application.

The Migration Level 0 allocation total shows the amount of DASD that was originally allocated by the data sets before they were migrated from Level 0.

The Backup Level 0 allocation total shows the amount of DASD that was allocated by the data sets when they were backed up from Level 0. The total is only accumulated for data sets that no longer exist at the Level 0 and provides you with an estimate of the amount of DASD that would be needed if all of the backup data sets were to be recovered at one time.

The Migration Level 1 and Level 2 data totals show the total size in KB of the migrated data set that currently resides at migration Levels 1 and 2. If compaction is in use, this value represents the compacted size.

The backup totals show the total size in KB of the backup data sets. If compaction is in use, this value represents the compacted size.

Figure 5-5 DFHSM Migration and Backup Totals

```

DFHSM MIGRATION AND BACKUP TOTALSVERSION 7.2
REPORT DATE: 12/31/2002                UNITS : KBYTES                PAGE : 1
----- MIGRATION TOTALS -----                ----- BACKUP TOTALS -----
--- ACCOUNT CODE ---
      LEVEL 0      LEVEL 1      LEVEL 2      LEVEL 0      BACKUP
      ALLOCATION    DATA      DATA      ALLOCATION    DATA
BATCH  195,237.25K  21,741.59K  1,342.62K   967.48K     101,296.03K
PROD   217,634.98K  33,010.24K  21,305.07K  31,895.36K  52,090.99K
TEST   131,467.50K  40,298.60K  19,281.89K   460.35K     51,050.55K
ONLINE 146,803.02K  28,452.00K   8,529.89K   67,457.45K  62,050.08K

```

SAS Reporting Procedures

Several SAS procedures are located in the SGCSAS01 member of the *?prefix.BBSAMP* library. These procedures can produce the following reports:

- tabular report of all space that is budgeted, by type with totals
- bar graph report of each application by type
- sorted list of all space that is budgeted, all active applications, and all current totals
- list of the 25 largest budget applications (in descending order)
- sorted list of all budget applications that exceed their established warning levels

Another SAS procedure, located in the SGCSAS02 member of the *?prefix.BBSAMP* library, maintains a trend analysis file for use as input to a subsequent SAS procedure (SGCSAS03) that produces a graphical representation of budget application growth trends.

Note: Output from the SAS reports is the same as the COBOL report output. For report samples, refer to Figure 5-6 on page 5-14 through Figure 5-12 on page 5-17.

COBOL Reporting Programs

The sample COBOL programs have been tested using the COBOL/II compiler. Sample JCL for compile and link edit is provided in the *?prefix.BBSAMP* library with member names SGC1CMLK, SGC2CMLK, and SGC3CMLK. These correspond to the source code SGCCOB01, SGCCOB02, and SGCCOB03. The JCL we provide must be customized to conform to your installation standards, your choice of data set names, and your compile procedure. (COB2UC is used in the samples.)

Reporting with the SGCCOB01 Member

- Report 1—List of all active applications, current allocation amounts, and budgets.
- Report 2 —List of up to 25 of the largest users of permanent and VSAM space (in descending order).
- Report 3—List of all applications over their warning threshold. The current allocation amounts, budgets, and warning percentages are listed for each entry over the warning level. A comment flag marks the entries that have exceeded their budget.
- Report 4—Summary report of the high-water marks associated with a user (USERFn) field. The user field is selected by the REPORT4 specifying REPORT4=1 | 2 | 3 in the parameter field to select the required user field. The default value is 1 (USERF1). You can suppress this report by specifying REPORT4=N in the parameter field.
- Report 5—Graph of current allocation amounts versus budgets for selected applications. The current amount is represented with an A, the high-water mark with an H, and the budget with an M. This report gives a graphical representation of how each application stands relative to its budget.
- Report 6—Alphabetic listing of applications.

Note: All reports listed above (with the exception of Report 4) can be selected or suppressed by specifying Y or N next to the report name in the parameter string for SGCCOB01. Samples of these reports follow.

A sample execution JCL for this COBOL reporting program can be found in member SGCCOBJ1 of the *?prefix.BBSAMP* library.

Figure 5-6 Database Detail Report

```

REPORT 1                                APPLICATION DATABASE DETAIL REPORT
VERSION 7.2
REPORT DATE : 12/31/2002 PAGE : 1

ACCOUNT/TYPE      CURRENT      BUDGET      HIGH WATER      NAME
USER FIELDS

CONTROL                                11/08/2002 BUDGET-DASD DEFAULT
RESYNC
PERMANENT          0          0          0 KBYTES
TEMPORARY          0          0          0 KBYTES
VSAM               0          0          0 KBYTES
HSM               0          0          0 KBYTES

ACCTNG                                11/17/2002 12/31/2001 ACCOUNTING/EDP
RESYNC
PERMANENT        36,935      100,000      66,432 KBYTES
TEMPORARY         0          10,000         0 KBYTES
VSAM             13,345       25,000      14,756 KBYTES
HSM             10,267       11,638      11,638 KBYTES

DEVLPMT                                11/21/2002 12/31/2001 PROJECT
DEVELOPMENT                                RESYNC
PERMANENT        13,949       65,000      24,187 KBYTES
TEMPORARY         0          1,000         0 KBYTES
VSAM             5,005       4,500       5,005 KBYTES
HSM             3,264       6,032       6,032 KBYTES

MISC 11/08/2002 12/28/2001 NON CLASSED DATA                                RESYNC
PERMANENT         1,390       2,500       1,689 KBYTES
TEMPORARY         0          500         0 KBYTES
VSAM             741       1,500       2,342 KBYTES
HSM             342       1,255       1,255 KBYTES

ONLINE                                11/08/2002 12/31/2001 ONLINE CICS
RESYNC
PERMANENT        198,741     999,999     198,741 KBYTES
VSAM INCLUDED IN PERMANENT
HSM INCLUDED IN PERMANENT

PROD                                11/08/2002 12/31/2001 PRODUCTION
APPLICATIONS                                RESYNC
PERMANENT        98,992     250,000     101,745 KBYTES
TEMPORARY         0          50,000         0 KBYTES
VSAM            103,342     150,000     159,641 KBYTES
HSM             99,331     100,438     100,438 KBYTES

SALES                                12/01/2002 12/29/2001 SALES ANALYSIS
RESYNC
PERMANENT        145,565     130,000     145,565 KBYTES
TEMPORARY         0          45,000         0 KBYTES
VSAM             17,517     12,500     19,602 KBYTES
HSM             10,344     11,498     11,498 KBYTES

TEST                                11/30/2002 12/30/2001 TEST APPLICATIONS
RESYNC
PERMANENT        37,183     50,000     52,905 KBYTES
TEMPORARY         0          9,000         0 KBYTES
VSAM             56,997     65,000     63,103 KBYTES
HSM             34,278     12,365     12,365 KBYTES

```

Figure 5-7 Top Five Users of Permanent and VSAM Space

REPORT 2 TOP 5 USERS OF PERMANENT AND VSAM SPACE
 VERSION 7.2
 REPORT DATE : 12/31/2002 PAGE : 1

ACCOUNT UNITS	PERMANENT			VSAM		
	CURRENT	BUDGET	HIGH WATER	CURRENT	BUDGET	HIGH WATER
PROD KBYTES	98,992	250,000	101,745	103,342	150,000	159,641
ONLINE KBYTES	198,741	999,999	198,741	0	0	0
SALES KBYTES	145,565	130,000	145,565	17,517	12,500	19,602
TEST KBYTES	37,183	50,000	52,905	56,997	65,000	63,103
ACCTNG KBYTES	36,935	100,000	66,432	13,345	25,000	14,756
TOTAL TOP KBYTES	517,416	1,529,999	455,366	191,201	252,500	257,102
TOTAL ALL KBYTES	532,755	1,597,499	455,366	196,947	258,500	264,449

Figure 5-8 Applications over Warning Threshold Report

REPORT 3 APPLICATIONS OVER WARNING THRESHOLD
 VERSION 7.2
 REPORT DATE : 01/31/2002 UNITS : KBYTES PAGE : 1

% ACCOUNT WARN	PERMANENT		TEMPORARY		VSAM	
	BUDGET COMMENT	CURRENT	BUDGET	CURRENT	BUDGET	CURRENT
DEVLPMT 80	65,000	13,949	1,000	0	4,500	5,005
SALES 80 ** PERM **	130,000	145,565	45,000	0	12,500	17,517

Trend Analysis Reporting

The second COBOL program, located in the SGCCOB02 member of the Installation Control library, maintains a trend analysis file for use as input to the third COBOL program. This third program, stored in the SGCCOB03 member of the *?prefix*.BBSAMP library, produces a tabular representation of application growth trends over the last 12 months. Sample execution JCL can be found in members SGCCOJB2 and SGCCOJB3 of *?prefix*.BBSAMP. Figure 5-12 shows a sample trend analysis report.

Note: For Application Trending purposes, MAINVIEW SRM retrieves start and end dates and times from the parameters in the TIME command. If you have not set the TIME command, the default is a duration of 7 days (168 hours) before the current date and time. For information about using the TIME command, see *Using MAINVIEW* or type Help TIME on the command line to access online help.

Figure 5-12 Database Trend Report

APPLICATION DATABASE TREND REPORT												VERSION 7.2											
REPORT DATE : 12/31/2002																							
												NUMBERS REPRESENT THE AMOUNT OF DASD USED IN KEYTES											
TYPE	JAN2002	FEB2002	MAR2002	APR2002	MAY2002	JUN2002	JUL2002	AUG2002	SEP2002	OCT2002	NOV2002	DEC2002											
ACCOUNT: ACCTNG																							
CUR-PERM	24,671	26,205	28,122	31,156	30,830	32,543	33,913	35,872	38,452	35,543	35,943	36,935											
HWM-PERM	24,671	27,105	28,455	31,156	31,156	32,978	33,913	36,456	38,501	38,501	66,432	66,432											
CUR-VSAM	10,205	11,437	10,386	12,365	13,187	12,675	11,873	12,904	14,002	14,009	13,646	13,345											
HWM-VSAM	11,403	11,437	11,437	12,365	13,845	13,845	13,977	13,987	14,002	14,009	14,009	14,756											
ACCOUNT: CONTROL																							
ACCOUNT: DEVLPMT																							
CUR-PERM						8,231	9,408	19,127	11,436	11,365	12,521	13,949											
HWM-PERM						8,231	9,408	24,187	24,187	24,187	24,187	24,187											
CUR-VSAM						3,690	3,690	3,690	5,005	5,005	5,005	5,005											
HWM-VSAM						3,690	3,690	3,690	5,005	5,005	5,005	5,005											
ACCOUNT: MISC																							
CUR-PERM	1,081	1,114	1,122	1,156	733	1,342	1,231	1,689	1,451	1,345	1,523	1,390											
HWM-PERM	1,081	1,114	1,321	1,321	1,321	1,342	1,342	1,689	1,689	1,689	1,689	1,689											
CUR-VSAM	405	723	809	914	1,145	923	1,363	1,559	1,007	909	646	741											
HWM-VSAM	703	824	824	914	1,234	1,234	1,477	1,559	1,559	1,559	2,342	2,342											
ACCOUNT: ONLINE																							
CUR-PERM	178,801	181,511	182,341	182,555	183,377	188,212	189,903	191,398	191,815	194,342	196,321	198,741											
HWM-PERM	178,801	181,511	182,341	182,555	183,377	188,212	189,903	191,398	191,815	194,342	196,321	198,741											
ACCOUNT: PROD																							
CUR-PERM	77,199	82,769	81,628	83,389	84,343	81,331	85,302	85,932	87,324	91,342	94,233	98,992											
HWM-PERM	77,199	82,769	82,769	83,389	84,567	84,567	85,532	86,229	90,329	93,324	101,745	101,745											
CUR-VSAM	92,132	92,429	96,393	95,331	95,215	94,292	96,789	145,839	159,641	132,893	121,825	103,342											
HWM-VSAM	92,132	93,244	96,393	96,393	96,393	96,393	96,789	145,839	159,641	159,641	159,641	159,641											
ACCOUNT: SALES																							
CUR-PERM												145,565	145,565										
HWM-PERM												145,565	145,565										
CUR-VSAM												16,543	17,517										
HWM-VSAM												17,364	19,602										
ACCOUNT: TEST																							
CUR-PERM												32,541	37,183										
HWM-PERM												52,905	52,905										
CUR-VSAM												51,847	56,997										
HWM-VSAM												55,735	63,101										

TSO Commands

The application collector is controlled by the following two TSO commands:

- BUDDSN command lists the applications associated with a data set.
- BUDGET command lists, queries, and modifies applications.

During the implementation process, the system administrator can use the BUDGET command to define the applications for your installation. The DASD space administrators may use the BUDGET command to define the DASD budgets for users within their groups.

Since the BUDGET and BUDDSN TSO commands may be available to all users, consider using the Security Exit to control access to the various functions of these commands. See Appendix A, “System Exits” for more information about exits.

When either of these TSO commands is issued, the command processor determines whether the file name SGCDB has been allocated to your TSO session. When allocated, the commands use this locally allocated application collector database. Otherwise, the commands use the active database.

BUDDSN Command

The BUDDSN command enables you to list the application codes associated with a disk data set. This command scans the entries in the VTOC and determines which application codes are associated with each data set. This TSO command can be used to list the application codes for a single data set, all cataloged data sets that match a pattern, or all data sets on a volume.

Listing Application Codes for a Single Data Set

The BUDDSN command can list the Application Collector application codes associated with a single data set. If the data set is cataloged, only the data set name is required. If the data set is not cataloged, the VOLSER parameter must be specified. To list the applications for a single data set, use the following format:

```
BUDDSN dataset VOLSER(volume-serial)
```

dataset

Positional parameter that specifies the fully qualified data set name to be listed. If the data set name is not enclosed in single quotation marks, TSO adds the profile prefix.

VOLSER(volume-serial)

Optional parameter that specifies the volume serial of an uncataloged data set. If not specified, the BUDDSN command locates the volume by way of the catalog entry. This parameter is required when listing uncataloged data sets.

To list the applications associated with the data set EMP.ABC.LOAD, type the following command:

BUDDSN 'EMP.ABC.LOAD'

The output should look similar to the following:

```
SGC3303  DSNAME=EMP.ABC.LOAD  VOLSER=MVS003
SGC3304  ---ACCOUNT=SYSTEMS  LEVEL=1
SGC3304  ---ACCOUNT=EMP  LEVEL=2
```

Listing Application Codes for a Group of Data Sets

The BUDDSN command also can list the applications associated with a range of cataloged data sets. The data set range is determined by a pattern given to catalog management.

To list all of the data sets using a pattern, use the following format:

```
BUDDSN pattern
```

pattern—Positional parameter that specifies a partial data set name followed by an asterisk. An asterisk cannot be embedded in the pattern; the asterisk must be the last character in the pattern.

Note: If the BUDDSN command is issued without parameters, all data set names for your TSO profile prefix are listed.

To list the application codes associated with the data sets beginning with EMP.ABC, type the following command:

```
BUDDSN 'EMP.ABC.*'
```

The output should look similar to the following:

```
SGC3303 DSNAME=EMP.ABC.MACLIB VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=EMP LEVEL=2
SGC3303 DSNAME=EMP.ABC.INSTALL VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=EMP LEVEL=2
SGC3303 DSNAME=EMP.ABC.LOAD VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=EMP LEVEL=2
```

Listing Application Codes for All Data Sets on a Volume

The third use of the BUDDSN command enables you to examine the applications associated with each of the data sets on a specific volume or set of volumes. This command scans the VTOC and lists each data set and the associated applications. If a volume has an indexed VTOC, the data sets are listed in alphabetic order. To list the applications associated with each of the data sets on a volume or set of volumes, use the following format:

```
BUDDSN * VOLSER(volume-serial)
```

*

When the BUDDSN command encounters an asterisk as the first positional parameter, all data sets found on the specified volume are listed.

VOLSER(*volume-serial*)

Specifies a volume serial or a partial volume serial followed by an asterisk. This is required when the first positional parameter is an asterisk.

Example

To list all of the data sets and their associated applications on volume MVS002, type the following command:

```
BUDDSN * VOLSER(MVS002)
```

The output should look similar to the following:

```
SGC3303 DSNAME=DEV.BAPROD.COBOL VOLSER=MVS002
SGC3304 ---ACCOUNT=APPLIC LEVEL=1
SGC3304 ---ACCOUNT=DEV LEVEL=2
SGC3303 DSNAME=DEV.BATST.COBOL VOLSER=MVS002
SGC3304 ---ACCOUNT=APPLIC LEVEL=1
SGC3304 ---ACCOUNT=DEV LEVEL=2
SGC3303 DSNAME=EMP.SGC.MACLIB VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=EMP LEVEL=2
SGC3303 DSNAME=SYS3.ABC.INSTALL VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=SYS3 LEVEL=2
SGC3303 DSNAME=SYS3.ABC.LOAD VOLSER=MVS002
SGC3304 ---ACCOUNT=SYSTEMS LEVEL=1
SGC3304 ---ACCOUNT=SYS3 LEVEL=2
```

BUDGET Command

The TSO BUDGET command can list, query, add, and change application information. The list feature enables you to examine a single application or a range of applications. Your installation may elect to use the security exit provided to restrict the modifying and reviewing of application information. Consider including this command in the logon CLIST to inform your users of their current DASD budget status.

The TSO BUDGET command selects applications by specifying an application code or by using a query. For example, a TSO BUDGET command can be used for the following tasks:

- to display the total space utilization of all applications whose application code that begins with ACCTNG. This report shows how much space the accounting department is currently using.
- to display the application codes used by the accounting department.

Queries can select applications by comparing a named field with a value or by comparing the values in two fields in an application. For example, a query could select all of the applications that currently have more than 10,000KB allocated. Or, the current allocation totals for each application can be compared to the budgeted totals for each application to select applications that are over budget.

A number of options can then be used to specify how the selected information should be displayed. These options can display totals, all of the application information, or just the application codes of the selected applications.

Command Coding Rules

To use the BUDGET command, observe the following coding rules:

- Enclose in single quotation marks any parameters that contain embedded blanks or special characters, for example, (ACCT,EQ,'ABC*').
- When specifying commands, you can use masking when comparing against non-numeric fields. Refer to the *MAINVIEW SRM for OS/390 Implementation Guide* for information on masking.

Note: When using masking, the operator must be EQ.

Listing a Single Application

This use of the BUDGET command displays the current information about a specific application. To list a single application, use the following format:

```
BUDGET acct-code LIST(ALL | ACCT)
```

- **acct-code**—Specifies the application to be listed. Enclose application codes that contain embedded blanks in single quotation marks.
- **LIST(ALL | ACCT)**—Displays application information.

LIST(ALL) displays the date created, last used date, warning level, index level, current space used, maximum space used, and high-water mark for permanent, temporary, HSM, and VSAM space. The name and user fields are also listed if they contain information. The default value is ALL.

LIST(ACCT) displays only the application codes for the application selected.

Example

To list the budget application ACCTING, type the following command:

```
BUDGET ACCTING
```

The output should look similar to the following:

```
SGC3202 ACCTING
SGC3203 DATE-CREATED=2000/02/27 LAST-USED=2000/08/12 WARN= 0%
      INDEX= 235 WARNING
SGC3204 NAME ='ACCOUNTING GROUP      '
SGC3205 TYPE=PERM CUR= 7640.00K MAX= 1000.00K HWM= 9230.00K
SGC3205 TYPE=TEMP CUR= 340.00K MAX= 900.00K HWM= 4350.00K
SGC3205 TYPE=VSAM CUR= 5460.00K MAX= 800.00K HWM= 5460.00K
SGC3205 TYPE=HSM CUR= 10920.00K      HWM= 11420.00K
```

Querying Several Applications

The BUDGET command also can query the the Application Collector database. In this mode, the entire database is read. Applications can be selected by comparing a field in an application to a value or by comparing a field to a field.

Comparing a Field to a Value

To compare a field to a value, use the following format:

```
BUDGET (field1,operator,value) LIST(ALL | ACCT | NONE) TOTAL
```

- **field1**—Specifies the name of the field whose contents are to be compared. Select a name for field1 from Table 5-2 on page 5-25.
- **operator**—Can be any of the following operators: EQ (equal), NE (not equal), GT (greater than), LT (less than), GE (greater than or equal to), or LE (less than or equal to).
- **value**—Specifies the value to which the field1 subparameter is to be compared. Can be any numeric or alphanumeric value. Enclose the value in single quotation marks if it contains blanks or special characters. Masking may be used.
- **LIST(NONE)**—When LIST(NONE) is specified, nothing is displayed. This can be used to limit output during mass database changes.
- **TOTAL**—Causes total space information to be displayed for all of the selected applications; optional.

Examples

To list the application codes of all of the applications that currently have more than 1000 kilobytes of permanent storage allocated, type the following command:

```
BUDGET (CURPERM,GT,1000) LIST(ACCT)
```

The output should look similar to the following:

```
SGC3202 ACCTING
SGC3202 PAYROLL
SGC3202 ENGINER
SGC3202 PRODUCT
SGC3202 OPERATI
```

To list the total space information for all of the applications that have a value of US starting in the fourth position of user field 1, type the following command:

```
BUDGET (USERF1,EQ,%%US*) LIST(NONE) TOTAL
```

The output should look similar to the following:

```
SGC3202 **TOTAL**
SGC3203 DATE-CREATED=2000/02/27 LAST-USED=2000/08/12 WARN= 0% INDEX= 235 WARNING
SGC3205 TYPE=PERM CUR= 1501.87K MAX= 1286665.38K HWM= 234099.00K
SGC3205 TYPE=TEMP CUR= 0.00K MAX= 1222221.00K HWM= 6951.00K
SGC3205 TYPE=VSAM CUR= 2202.71K MAX= 1251109.86K HWM= 220271.00K
SGC3205 TYPE=HSM CUR= 3703.68 HWM= 174562
```

Comparing a Field to a Field

To compare the contents of a field to that of a second field in an application record, use the following format:

```
BUDGET (field1,operator,field2) LIST(ALL | ACCT | NONE) TOTAL
```

- **field2**—Specifies the field to which the field1 parameter is to be compared. Select the value for the field2 parameter from Table 5-2 on page 5-25.

Example

To list only the application codes for all applications where the current VSAM space is greater than the current permanent space:

```
BUDGET (CURVSAM,GT,CURPERM) LIST(ACCT)
```

The output should look similar to the following:

```
SGC3202 ACCTING
SGC3202 PRODUCT
SGC3202 OPERATI
```

Field Table for the BUDGET Command

Table 5-2 provides the field names and descriptions of the fields that you can use in field-to-value and field-to-field comparisons.

Table 5-2 Field Table for BUDGET Command

Field name	Description
CURPERM	Current permanent space allocation
CURTEMP	Current temporary space allocation
CURVSAM	Current VSAM space allocation
CURHSM	Current DFHSM space allocation
HWMPERM	High-water mark permanent space
HWMTEMP	High-water mark temporary space
HWMVSAM	High-water mark VSAM space
HWMHSM	High-water mark DFHSM space
ACCT	Application code
MAXP	Permanent space allocation budget
MAXPERM	Permanent space allocation budget
MAXT	Temporary space allocation budget
MAXTEMP	Temporary space allocation budget
MAXV	VSAM space allocation budget
MAXVSAM	VSAM space allocation budget
NAME	User name
USERF1	User field 1
USERF2	User field 2
USERF3	User field 3
USRF1	User field 1 (alias for USERF1)
USRF2	User field 2 (alias for USERF2)
USRF3	User field 3 (alias for USERF3)
WARN	Warning percentage

Adding or Changing a Single Application

This section describes how the BUDGET command adds or changes a single application. To add or change a single application, use the following format:

```

BUDGET acct-code ADD|CHANGE
      FROM(from-acct-cd)
      NAME(user-info)
      USERF1(user-info)
      USERF2(user-info)
      USERF3(user-info)
      MAXPERM(numeric)
      MAXTEMP(numeric) | NOTEMP | PTEMP | TTEMP
      MAXVSAM(numeric) | NOVSAM | PVSAM | TVSAM
      NOHSM | PHSM | THSM
      WARN(numeric < 100)
      RESET
      MONITOR | WARNING | REJECT | DEFAULT
      DELETE | NODELETE
      KILOBYTE | MEGABYTE | GIGABYTE | TERABYTE | PETABYTE | EXABYTE
      K1000 | K1024

```

- **acct-code**—Required positional parameter that specifies the application you want to change. Enclose in single quotation marks those application codes that contain blanks or special characters.
- **ADD | CHANGE**—Required positional parameter. Specify ADD to create a new entry; CHANGE to alter an existing entry.
- **FROM(from-acct-cd)**—Optional parameter that specifies the default or model application from which the following are copied: MAXPERM, MAXTEMP, MAXVSAM, WARN, NAME, and User fields. Parameters specified on the BUDGET command override values copied from the application.
- **NAME(user-info)**—Optional parameter in which you can specify up to 20 characters of information. Enclose the information in single quotation marks if it contains embedded blanks or special characters.
- **USERF1(user-info)**—Optional parameter in which you can specify up to eight characters of information. Enclose the information in single quotation marks if it contains embedded blanks or special characters.
- **USERF2(user-info)**—Optional parameter in which you can specify up to 10 characters of information. Enclose the information in single quotation marks if it contains embedded blanks or special characters.

- **USERF3(user info)**—Optional parameter in which you can specify up to 10 characters of information. Enclose the information in single quotation marks if it contains embedded blanks or special characters.
- **MAXPERM(numeric)**—Optional parameter that specifies the maximum amount of permanent space kilobytes (depending on the system option) for this application. If you specify the PTEMP and/or PVSAM parameters, the value specified for the MAXPERM parameter should also include the maximum temporary and VSAM space allocations.
- **MAXTEMP(numeric)**—Optional parameter that specifies the maximum amount of temporary space kilobytes for this application. TTEMP is not required if MAXTEMP is specified.
- **NOTEMP**—Optional parameter that directs the Application Collector to ignore all allocations for temporary space.
- **PTEMP**—Optional parameter that directs the Application Collector to consider temporary space the same as permanent space and include it in that total. If the PTEMP option is changed after temporary space has been allocated, it is *not* added into the total for permanent space, nor is it cleared.
- **TTEMP**—Optional parameter that directs the Application Collector to track temporary space. TTEMP is the default when adding an application.

Note: You can specify MAXTEMP, NOTEMP, PTEMP, or TTEMP. TTEMP is the default.

- **MAXVSAM(numeric)**—Optional parameter that specifies the maximum amount of VSAM space in tracks or kilobytes for this application. TVSAM is not required if MAXVSAM is specified.
- **NOVSAM**—Optional parameter that directs the Application Collector to ignore all allocations for VSAM space.
- **PVSAM**—Optional parameter that directs the Application Collector to consider VSAM space the same as permanent space and include it in that total. If the PVSAM option is changed after VSAM space has been allocated, it is *not* added into the total for permanent space, nor is it cleared.

- **TVSAM**—Optional parameter that directs the Application Collector to track VSAM space. TVSAM is the default when adding an application.

Note: You can specify MAXVSAM, NOVSAM, PVSAM, or TVSAM. TVSAM is the default.

- **NOHSM**—Optional parameter that directs the Application Collector to ignore all DADSM requests for DFHSM migrate or backup data sets. This option overrides the HSMTRACK (YES) system option.

- **PHSM**—Optional parameter that directs the Application Collector to consider DFHSM migrate and backup space the same as permanent space and include it in that total. If the PHSM option is changed after DFHSM migrate and backup space has been allocated, it is *not* added into the total for permanent space. This option has an effect only if the system option, HSMTRACK (YES) is specified.

- **THSM**—Optional parameter that directs the Application Collector to track HSM space. THSM is the default when adding an application.

Note: You can specify NOHSM, PHSM, or THSM.

- **WARN(n < 100)**—Optional parameter that specifies the percentage of the budget maximum for this type of space to be used before a warning message is generated. It must be an integer between 0 and 100.

- **RESET**—Optional parameter that causes all of the high-water marks to be set to the current allocation amounts.

- **MONITOR | WARNING | REJECT | DEFAULT**—Optional parameter that sets the application mode option. The application mode option overrides the mode specified as a system option.

— Applications running in MONITOR mode only record space allocation totals.

— If an application is running in WARNING mode, a warning message is generated if the budget is exceeded.

— If an application is running in REJECT mode, the budget amount is used to place a limit on the amount of space that can be allocated. Any allocation request that would cause an application to exceed its budget is denied.

— The DEFAULT mode specification can be used to clear the application mode option. The default (or system option) mode is used.

- **DELETE | NODELETE**—Optional parameter that sets the deletion-requested flag. Applications flagged for deletion can be dropped from the database by using the INIT FILE (UPDATE) function of SGCMAINT. NODELETE resets the deletion-request flag.

Note: The MAINVIEW SRM Reporting component uses budgets as the basis for percentage-usage calculations and graphic and tabular displays. BMC Software recommends that you specify a budget for all applications that run in monitor mode. This budget will be used only for calculation-reporting purposes.

- **KILOBYTE | MEGABYTE | GIGABYTE | TERABYTE | PETABYTE | EXABYTE** (aliases are **KB | MB | GB | TB | PB | EB** respectively)—Optional scaling parameter that determines the units of all numeric values entered. The default value is KILOBYTE.
- **K1000 | K1024**—Optional parameter that defines a kilobyte and higher powers for all input and output values, for example, a terabyte is 1000^4 if K1000 is specified and 1024^4 if K1024 is specified. The default value is derived from the definition in the KBYTECALC global parameter. For definitions of global parameters, see the *MAINVIEW SRM User Guide and Reference*.

Examples

The following examples show the commands you can use to add or change an application code.

- To add an application with an application code of ACCTA, type the following command:

```
BUDGET ACCTA ADD
```

- To add an application using application MODEL#1 as a model, type the following command:

```
BUDGET ACCTB ADD FROM(MODEL#1)
```

- To add an application overriding the value of NAME from the model, type the following command:

```
BUDGET ACCTC ADD FROM(MODEL#1)
NAME ('ACCOUNT C')
```

- To add an application overriding the value of MAXPERM from the default or CONTROL application, type the following command:

```
BUDGET ACCTD ADD MAXPERM(50000)
```

- To change the value of MAXVSAM for ACCTA, type the following command:

```
BUDGET ACCTA CHANGE MAXVSAM(60000)
```

- To set the deletion request flag for ACCTB, type the following command:
- BUDGET ACCTB CHANGE DELETE

Mass Changes

You can make changes to multiple applications by using the mass change function of the BUDGET command.

Warning! Use this function with extreme care to avoid making unintentional changes.

Before making mass changes, you may want to test the command syntax against a copy of the active database.

As with the LIST parameter, you can select applications by comparing a field to a value or a field to another field. To compare a field to a value, use the following format:

```

BUDGET (field1,operator,value)
CHANGE FROM(from-acct-cd)
        NAME(user-info)
        USERF1(user-info)
        USERF2(user-info)
        USERF3(user-info)
        MAXPERM(numeric)
        MAXTEMP(numeric) | NOTEMP | PTEMP | TTEMP
        MAXVSAM(numeric) | NOVSAM | PVSAM | TVSAM
        NOHSM | PHSM | THSM
        WARN(n < 100)
        RESET
        MONITOR | WARNING | REJECT | DEFAULT
        KILOBYTE | MEGABYTE | GIGABYTE | TERABYTE | PETABYTE | EXABYTE
        K1000 | K1024
        K1000 | K1024
    
```

Example

To change the maximum space a permanent data set can allocate for every budget application that contains a user field 2 of ACCTING, type the following command:

```
BUDGET (USERF2,EQ,'ACCTING') CHANGE MAXPERM(2000)
```

To compare a field to another field, use the following format:

```
BUDGET (field1,operator,field2)
CHANGE FROM(from-acct-cd)
NAME(user-info)
USERF1(user-info)
USERF2(user-info)
USERF3(user-info)
MAXPERM(numeric)
MAXTEMP(numeric) | NOTEMP | PTEMP | TTEMP
MAXVSAM(numeric) | NOVSAM | PVSAM | TVSAM
NOHSM | PHSM | THSM
WARN(n < 100)
RESET
MONITOR | WARNING | REJECT | DEFAULT
KILOBYTE | MEGABYTE | GIGABYTE | TERABYTE | PETABYTE | EXABYTE
K1000 | K1024
K1000 | K1024
```

Example

To set the warning percentage to 70 percent in all applications where user field2 matches user field3, type the following command:

```
BUDGET (USRF2,EQ,USRF3) CHANGE WARN(70)
```

Note: Several special functions of the BUDGET command allow authorized users to change current allocation amounts or high-water marks. Refer to Table 5-2 on page 5-25 for more information.

Special Functions of the TSO BUDGET Command

The BUDGET command can be used to change current allocation amounts or high-water marks. In general, you should never need to change the allocation amounts. The system provides the RESYNC program to rebuild the application database. These extensions to the TSO BUDGET command offer you the ability to change the allocation amounts without the overhead of a RESYNC.

Note: Closely monitor use of the following commands to avoid unintentional changes.

As with the standard functions, the BUDGET command can change a single application or be used to make mass changes. The format to change a single application follows.

BUDGET	acct-code	CHANGE
		CURPERM(numeric)
		CURTEMP(numeric)
		CURVSAM(numeric)
		CURHSM(numeric)
		HWMHSM(numeric)
		HWMPERM(numeric)
		HWMTEMP(numeric)
		HWMVSAM(numeric)

Command Parameters

- **acct-code**—Required parameter that specifies the budget application to be changed.
- **CURPERM(numeric)**—Optional parameter that changes the amount of space allocated to permanent data sets. This field may include temporary and VSAM space.
- **CURTEMP(numeric)**—Optional parameter that changes the amount of space allocated to temporary data sets.
- **CURVSAM(numeric)**—Optional parameter that changes the amount of space allocated to VSAM data sets.
- **CURHSM(numeric)**—Optional parameter that changes the amount of space allocated to DFHSM migrate and backup data sets.
- **HWMHSM(numeric)**—Optional parameter that changes the high-water mark for DFHSM migrate and backup data sets.

- **HWMPerm(numeric)**—Optional parameter that changes the high-water mark for permanent data sets.
- **HWMTemp(numeric)**—Optional parameter that changes the high-water mark for temporary data sets.
- **HWMVSAM(numeric)**—Optional parameter that changes the high-water mark for VSAM data sets.

Examples

The following examples demonstrate uses of the BUDGET command special functions. Note that parameters previously defined are not repeated here.

- To change the amount of space allocated to permanent data sets for application ACCTA, type the following command:

```
BUDGET ACCTA CHANGE CURPERM(123456)
```

- To update multiple applications, you can select applications by comparing a field to a value. The format used to compare a field to a value is as follows:

```
BUDGET (field1,operator,value)
      CHANGE  CURPERM(numeric)
           CURTEMP(numeric)
           CURVSAM(numeric)
           CURHSM(numeric)
           HWMPerm(numeric)
           HWMTemp(numeric)
           HWMVSAM(numeric)
           HWMHSM(numeric)
```

- To set the amount of permanent space to zero in all applications with an application code starting with ACCT, type the following command:

```
BUDGET (ACCT,EQ,'ACCT*') CHANGE CURPERM(0)
```

- You can also select applications by comparing the contents of two fields within an application using the following format:

```

BUDGET (field1,operator,field2)
      CHANGE  CURPERM(numeric)
            CURTEMP(numeric)
            CURVSAM(numeric)
            CURHSM(numeric)
            HWMPERM(numeric)
            HWMTEMP(numeric)
            HWMVSAM(numeric)
            HWMHSM(numeric)
    
```

- To set the amount of currently used VSAM space to zero in all applications where user field2 matches user field3, type the following command:

```
BUDGET (USRF2,EQ,USRF3) CHANGE CURVSAM(0)
```

Chapter 6 HSM Collector

This chapter presents the following topics:

How the HSM Collector Works	6-2
DFHSM Control Data Sets	6-2
DFHSM Log Extract Process	6-3
Automated Processing	6-4
Batch Processing	6-5
Log Extract Data Set	6-7
User Job Execution	6-7
Sysplex Environments with Shared DASD Considerations	6-8
System Parameters	6-8
DFHSM Output Management	6-14
Message Filtering	6-15
Input Data Sets	6-16
Output Data Sets	6-17
Skeleton Processing	6-28
DFHSM Output Management Views	6-34

How the HSM Collector Works

Information on the DFHSM environment is supplied to MAINVIEW SRM in several ways:

- The DFHSM control data sets are identified in EZHSM statements in the MAINVIEW SRM ISPF startup CLIST.
- Records from the DFHSM logs are extracted by MAINVIEW SRM under the control of a timer-activated task, which first switches the log file and then copies required records to a dynamically allocated MAINVIEW SRM data set.
- Specific information needed by MAINVIEW SRM, is supplied in parameters in the MAINVIEW SRM SMMSYSxx parmlib member.

These methods are discussed in detail in the following pages.

DFHSM Control Data Sets

The MAINVIEW SRM Reporting component derives information from the DFHSM MCDS and BCDS. These data sets are dynamically allocated during the start of the Reporting component. The DDnames are SRMMCDS, SRMBCDS, and SRMOCDS. The data set names used in the dynamic allocation are retrieved from the SMMSYSxx parmlib member as described in “System Parameters” on page 6-8.

The Reporting component does not support DFHSM control data sets that are defined as multi-cluster, key-range VSAM files. This type of control data set definition allows the control data sets to occupy multiple volumes for a total of 16 GB of space. Use of this type of control data set requires IBM APAR OY59526. Typically, only large installations use this type of control data set.

If you use multi-cluster key-range control data sets, do not use the Info line command for migrated data sets or the following views:

- HSMDLYA (Daily Activity Summary)
- HSMDLYV (Daily Volume Summary)

The Reporting component supports the following:

- control data sets that are single volume, single cluster (the most common organization) or multivolume, single cluster

- VSAM Record Level Sharing (RLS) for DFHSM

Using the RLS=Y SMMSYSxx parameter, the system determines how to open the Control Data Sets (CDSs) (see “System Parameters” on page 6-8).

- data sets that exceed four gigabytes in DFHSM

Note: RLS=Y is not required for RLS CDSs defined with share option 2. The IBM “share 2” fix allows non-RLS read access concurrent with RLS access. The share 2 fix was implemented with IBM PTF UW44099 (DFSMS 1.3) and UW44100 (DFSMS 1.4). The share 2 fix is installed on IBM operating systems higher than DFSMS 1.4.

DFHSM Log Extract Process

A number of the views are based on data in the DFHSM log files. This data must be extracted from the log files and written into a MAINVIEW SRM data set. DFHSM must be set to write data to its log file Y in order for the Reporting component to gather the extract data. The Reporting component does not use DFHSM PDA files for gathering data.

To reliably capture all of the DFHSM log data, a program is supplied that performs the following tasks:

- switches the DFHSM log file
- reads the newly inactive log file
- dynamically allocates an output file
- writes the data to the new file

Once started, this program is executed periodically, creating multiple date and timestamped extract files during a day. The Reporting component reads as many of these extract files as necessary to cover the time period requested for a view.

Tip: You can run the log-extract program as part of MAINVIEW SRM or in batch mode.

Automated Processing

The log-extract program is started at MAINVIEW SRM startup (or during MAINVIEW SRM execution) when the SMMSYS xx parameter HLOGCOLL is set to YES. You can set HLOGCOLL to NO to deactivate this program. If HLOGCOLL is set to NO, the only option for collecting log data is through batch execution.

As distributed on the MAINVIEW SRM installation tape, HLOGCOLL is set to NO. You can change it to YES in the MAINVIEW SRM System Parameters panel while MAINVIEW SRM is active to start log file processing; stopping and starting MAINVIEW SRM is not required.

Note: The HLOGCOLL parameter must still be changed in member SMMSYS xx to ensure that log file processing is activated on the next startup of MAINVIEW SRM.

HLOGYDSN must also be set in the SMMSYS xx member so that the system knows what DFHSM is using as its log file Y.

Use the HLOGAUTH (hours) and HLOGAUTM (minutes) parameters in SMMSYS xx to run the log-extract program at periodic intervals. Set intervals sufficiently short to ensure that the DFHSM log file does not fill up and get switched by DFHSM; if this happens, that log file data is lost to the views. Do not set any interval shorter than necessary, however, because of the following reasons:

- the number of input data sets is limited to 255
- the time required to allocate multiple data sets dynamically can significantly slow view generation

Warning! The MAINVIEW SRM Reporting component does not detect a log switch that occurs between the intervals specified. As a result, if DFHSM does a log swap between intervals, the system does not capture this log data for reporting. BMC Software recommends batch processing to prevent any missing data from views.

Batch Processing

Note: The *MAINVIEW SRM User Guide and Reference* provides batch processing procedures for accessing information that is found in online views.

You can run the log-extract program in batch mode. This may be preferable at your installation if you already have some type of automated operations software that can detect when DFHSM switches the log file; for example, a program could detect message ARC0020.

HLOGCOLL must be set to NO to run the batch version of the log-extract program. MAINVIEW SRM must be running. You can run the log-extract program in two modes: automatic and single execution. In automatic mode, the program swaps the log file and extracts data at regular intervals. In single-execution mode, the program is executed once and ends. You can also run it without swapping the DFHSM log file.

The following parameters are available for batch mode processing.

- NOAUTO—The log-extract program does not run automatically. The batch execution ends after processing the log file. If NOAUTO is not specified, the log-extract program continues to run until cancelled (or until MAINVIEW SRM is shut down), processing the log file on the interval specified by HLOGAUTH/M.
- NOSWAP—The log-extract program does not swap the DFHSM log file. It reads the current DFHSM Y log file. If NOSWAP is not specified, the log-extract program swaps the DFHSM log file prior to reading and extracting log records.

If the log-extract program is running in automatic mode, the log-extract program is terminated if HLOGCOLL is set to YES in MAINVIEW SRM; the MAINVIEW SRM-controlled version continues the processing.

If both the NOAUTO and NOSWAP parameters are specified, the log-extract program can be used to extract log file data from any desired log file of any name. This feature is useful, for example, to extract data from DFHSM log files on different LPARs than the one where MAINVIEW SRM is executing.

When 'NOAUTO,NOSWAP' is specified, the log-extension program scans the job JCL for any DD that begins with HLOG. If one is found, that data set is opened as the Y log file; the log file name in the SMMSYSxx member is ignored.

Note: Even if NOAUTO and NOSWAP are specified, a non-zero HLOGAUTH value *must* be specified

Note the following additional batch processing considerations:

- The last four characters of the DDname are taken as the system ID. This system ID should match the system that generated the log file; if it is different, views selected for that system ID are not complete.

Note: The value specified for the system ID portion of the HLOG DDname is inserted in the type-3 log records in the file. The value is used to form the low-level qualifier or the log extract data set name.

If you specify an incorrect system ID, you may see this incorrect value in views and be unable to select statistical and error views based on system ID.

- If only HLOG is specified, the current system ID of the system where the log-extract program is running is used.
- If the job JCL does not have a DD that begins with HLOG, the DFHSM Y log file named in SMMSYSxx is used.
- Only one log file is read per execution of the log-extract program.

The JCL used to execute the log-extract program in batch mode is in *?prefix*.BBSAMP member JCLHSMXLX. Modify JCLHSMXLX according to the instructions in the member header.

The example shown in Figure 6-1 extracts data from two log files.

Figure 6-1 Example JCL that Extracts Data from Two Log Files

```
//JOB
//STEP10 EXEC PGM=PRSANP01, PARM='NOSWAP,NOAUTO'
//STEPLIB DD DSN=EMP.MVSRM.BBLINK, DISP=SHR
//HLOGMVSA DD DSN=DFHSM.Y.LOGFILE.MVSA, DISP=SHR
//STEP20 EXEC PGM=PRSANP01, PARM='NOSWAP,NOAUTO'
//STEPLIB DD DSN=EMP.MVSRM.BBLINK, DISP=SHR
//HLOGMVSB DD DSN=DFHSM.Y.LOGFILE.MVSB, DISP=SHR
```

Starting DFHSM with LOGSW=YES causes a log swap by DFHSM at start up. To ensure that the data in the startup swapped log file is extracted for use, you should add a step to the DFHSM procedure preceding the DFHSM start to run the log extraction program with PARM='NOAUTO,NOSWAP'.

Log Extract Data Set

The log-file-switching program generates a data set that is named according to the following format:

user-specified-prefix.Dyymmdd.Thhmmss.SYSsyst-id

The variables in the data set name are defined as follows:

- *user-specified-prefix* is specified in the HLOGINDX parameter and can be up to 20 characters long. The prefix must be valid for data set naming restrictions, as in the following example:

```
BMC.DFHSMLOG.D941210.T225212.SYSMVS1
```

- *Dyymmdd* is the date specified as year, month, and day.
- *Thhmmss* is the time specified as the hour, minute, and second.
- *syst-id* is the system ID for the system on which the log-extract program is run.

Use the HLOGINDX parameter of the SMMSYS_{xx} parmlib member to control the data set name prefix.

The MAINVIEW SRM log extract file is allocated by default with primary space of 15 tracks and secondary space of 10 tracks. If this is insufficient for the amount of data collected in the interval specified in the HLOG_{xxxx} parameters, use the HLOGPRIM parameter to specify the size of the primary extent in tracks. The size of the secondary extent is set to one-half the size of the primary extent. Alternatively, you can use the MAINVIEW SRM function SPACSQTY to increase the primary and secondary allocation sizes.

User Job Execution

You can use HLOGTASK to run a procedure associated with the logfile switch that MAINVIEW SRM performs. HLOGTASK specifies the name of the procedure to be run.

Sysplex Environments with Shared DASD Considerations

The HLOGINDX parameter must be unique to each system on which HSM log collections are being performed. The Service Point for each Reporting component must be accessible to a CAS within the sysplex. Each CAS must in turn be accessible from other CASs within the sysplex. When running reports, use the CONTEXT command in MAINVIEW to view log data on other systems. MAINVIEW will handle the appropriate retrieval of data between the systems.

System Parameters

Table 6-1 lists the parameters in the SMMSYSxx PARMLIB member that are used to define and control log file support.

Table 6-1 System Parameters

Parameter	Required	Description
BCDSn=xxxxxxxxxxx	No	specifies backup control data sets to be defined and allocated during system startup
MCDSn=xxxxxxxxxxx	No	specifies migration control data sets to be defined and allocated during system startup
OCDS=xxxxxxxxxxx	No	specifies OCDS data set to be defined and allocated during system startup
HLOGAUTH=nn	Yes	automatic DFHSM log switch interval in hours
HLOGAUTM=nn	No	automatic DFHSM log switch interval in minutes
HLOGCOLL= YES/NO	No	activation of DFHSM log file data collection
HLOGINDX=xxxxxxxxx	No	DSN prefix of DFHSM log extract file
HLOGLIM=xxxx	No	limits the number of log extract data sets that are allocated for processing for one report request
HLOGPRIM=nnnn	No	number of tracks of primary allocation of log extract file
HLOGTASK=xxxxxxxxx	No	name of the procedure to be run after the DFHSM log file switch
HLOGUNIT=xxxxxxxxx	No	unit name for allocation of log extract file
HLOGYDSN=xxxxxxxxx	No	data set name of DFHSM's log file Y
RLS= Y/N	No	specifies whether the system should open the HSM Control Data Sets (CDSs) in VSAM Record Level Sharing mode

BCDS n =

Purpose: Specifies backup control data sets to be defined and allocated during system startup.

Syntax: BCDS n =xxxxxxxx

The variable n is the multi-cluster number. Specify the numbers 2, 3, and 4 only if you are defining a multi-cluster data set.

The variable xxxxxxxx is a backup data set name.

Required: No. If not specified MAINVIEW SRM will discover the HSM BCDS data set name when started. BMC Software recommends that you let the data set names be discovered; however, discovery requires that the DFHSM address space be active prior to SVOS being started. If this requirement cannot be met, this keyword should be used to specify the name.

Under special circumstances it is desired to not allocate any of the HSM CDS data sets. When this is the situation, any one of the CDS files must be specified as 'NONE'. All of the CDS data sets (BCDS1, MCDS1, and OCDS) may be specified as 'NONE' but it is only necessary to specify one for this feature to take place. In this situation some views will not be available.

Default: None

MCDS n =

Purpose: Specifies migration control data sets to be allocated during system startup.

Syntax: MCDS n =xxxxxxxx

The variable n is the multi-cluster number. Specify the numbers 2, 3, and 4 only if you are defining a multi-cluster data set.

The variable xxxxxxxx is a migrated data set name.

Required: No. If not specified MAINVIEW SRM will discover the HSM MCDS data set name when started. BMC Software recommends that you let the data set names be discovered; however, discovery requires that the DFHSM address space be active prior to SVOS being started. If this requirement cannot be met, this keyword should be used to specify the name.

Under special circumstances it is desired to not allocate any of the HSM CDS data sets. When this is the situation, any one of the CDS files must be specified as 'NONE'. All of the CDS data sets (BCDS1, MCDS1, and OCDS) may be specified as 'NONE' but it is only necessary to specify one for this feature to take place. In this situation some views will not be available

Default: None

OCDS=

Purpose: Specifies offline CDS data sets to be defined and allocated during system startup.

Syntax: OCDS=*xxxxxxx*

The variable xxxxxxx is an OCDS data set name.

Required: No. If not specified MAINVIEW SRM will discover the HSM OCDS data set name when started. BMC Software recommends that you let the data set names be discovered; however, discovery requires that the DFHSM address space be active prior to SVOS being started. If this requirement cannot be met, this keyword should be used to specify the name.

Under special circumstances it is desired to not allocate any of the HSM CDS data sets. When this is the situation, any one of the CDS files must be specified as 'NONE'. All of the CDS data sets (BCDS1, MCDS1, and OCDS) may be specified as 'NONE' but it is only necessary to specify one for this feature to take place. In this situation some views will not be available.

Default: None

HLOGAUTH=

Purpose: Specifies the *hours* component of the duration between automatic log file switching.

The Reporting component extracts records from the DFHSM log files. If HLOGCOLL=YES is specified, MAINVIEW SRM automatically switches the DFHSM log file and extracts the required records. The switching interval is specified in hours and minutes by the parameters HLOGAUTH and HLOGAUTM.

Syntax: HLOGAUTH=*nn*

The variable *nn* specifies a number of hours in the range of 0 through 24.

Required: Yes

Default: HLOGAUTH=00

HLOGAUTM=

Purpose: Specifies the *minutes* component of the duration between automatic log file switching.

The Reporting component extracts records from the DFHSM log files. If HLOGCOLL=YES is specified, MAINVIEW SRM automatically switches the DFHSM log file and extracts the required records. The switching interval is specified in hours and minutes by the parameters HLOGAUTH and HLOGAUTM.

Syntax: HLOGAUTM=*nn*

The variable *nn* specifies a number of minutes in the range of 0 through 60.

Required: No

Default: HLOGAUTM=00

HLOGCOLL=

Purpose: Specifies whether MAINVIEW SRM automatically performs DFHSM log file switching and record extraction.

The Reporting component requires certain records from the DFHSM log files. If HLOGCOLL=YES is specified, MAINVIEW SRM automatically switches the DFHSM log files and runs the log-extract program at the interval specified by the HLOGAUTH/M parameters.

Syntax: HLOGCOLL=*YES/NO*

Required: No

Default: HLOGCOLL=NO

HLOGINDX=

Purpose: Specifies the prefix of the data set that contains the records extracted from the DFHSM log file. The prefix can contain up to 20 characters in any number of name qualifiers. The full data set name generated for the log extract file is

prefix.Dyymmdd.Thhmmss.SYSsystem-id

Syntax: HLOGINDX=xxxxxxxxxxxxxxxxxx

Required: No

Default: None

HLOGLIM=

Purpose: Specifies a limit to the number of log extract data sets that will be allocated for one report request. By default, the system calculates the number of log extract data sets that it will allocate based upon how many active allocations reside in the SVOS address space. This parameter enables users to limit the number of log extracts that get allocated for one request so that one request will not use up all of the available allocation openings, which will cause other requests to prematurely end without gathering all of the requested data and lower the amount of I/O used by the SVOS address space.

Syntax: HLOGLIM=xxx

where xxx is a number in the range of 0–3000

Required: No

Default: none

HLOGPRIM=

Purpose: Specifies the number of tracks to be allocated as the primary extent amount for the log extract file. One-half of the primary extent is allocated for the secondary extent (with a minimum of 1). If not specified, 15 tracks are used for the primary extent, and 10 tracks are used for the secondary extent.

Syntax: HLOGPRIM=nnnn

The variable *nnnn* is a number in the range of 1 through 9999.

Required: No

Default: HLOGPRIM=15

HLOGTASK=

Purpose: Specifies the name of the procedure to be run after the DFHSM log file switch program execution.

The Reporting component extracts records from the DFHSM log files. If HLOGCOLL=YES is specified, MAINVIEW SRM automatically switches the DFHSM log file and extracts the required records.

You can use HLOGTASK to run a procedure associated with the logfile switch that MAINVIEW SRM performs.

Syntax: HLOGTASK=xxxxxxx

The variable xxxxxxx represents 1 to 8 characters.

Required: No

Default: None

HLOGUNIT=

Purpose: Specifies the esoteric or generic unit name for the allocation of the log extract file. If not specified, SYSALLDA is used.

Syntax: HLOGUNIT=xxxxxxx

Required: No

Default: HLOGUNIT=SYSALLDA

HLOGYDSN=

Purpose: Specifies the fully qualified data set name of the DFHSM log file Y.

Syntax: HLOGYDSN=xxxxxxxx.xxxxxxxxx...

Required: No. If not specified MAINVIEW SRM will discover the HSM LOG Y data set name when started. BMC Software recommends that you let the data set names be discovered; however, discovery requires that the DFHSM address space be active prior to SVOS being started. If this requirement cannot be met, this keyword should be used to specify the name.

Default: None

RLS=

Purpose: Specifies whether the system should open the HSM Control Data Sets (CDSs) in VSAM Record Level Sharing mode available for DFHSM.

Syntax: RLS=Y/N

Required: N

Default RLS=N

DFHSM Output Management

DFHSM output management allows automated processing of DFHSM and DFDS output. Output management provides the following benefits:

- filtering of unwanted or informational messages, displaying only messages that indicate situations that need attention
- automated creation of control cards, JCL, or commands to handle conditions described by messages, using information extracted from the message text
- immediate submission of generated JCL
- online views produced by the message-filtering process

Output management begins with the execution of a batch job. It reads user-specified sets of DFHSM logs and DFDSS message output. You can select messages for inclusion in reports and for construction of JCL or control cards. You can view the results online in the HSMOMDS and HSMOMML views.

Tip: Your DFHSM initialization parameters must include ACTLOGTYPE(DASD). DFHSM initialization parameters are usually stored in the ARCCMDxx member of your SYS1.PARMLIB data set. This logging method allows your log data to be available for the reporting facility.

Message Filtering

DFHSM and DFDSS produce one or more messages for every action they take. In some cases, they also produce messages for actions they do not take. This results in a large number of messages that might contain important information for managing the installation's storage resources. The storage administrator should regularly scan the messages to determine if any preventive or corrective action is needed.

For example, the following message indicates that a backup was not taken; the storage administrator should investigate and correct the situation. However, this message is often buried in many occurrences of messages ARC0722I, ARC0723I, and ARC0734I.

```
ARC0706I BACKUP OF VOLUME WORK02 TERMINATED, NO DASD
        DAILY BACKUP VOLUME AVAILABLE
```

```
ARC0700I BACKUP OF VOLUME SMS005 SUSPENDED, TASK WAITING
        FOR AN AVAILABLE DASD DAILY BACKUP VOLUME
```

DFHSM log data sets typically contain hundreds of ARC0734I messages, most of them indicating successful actions. However, the following message could indicate DFHSM processing bottlenecks that should be corrected.

```
ARC0734I ACTION=BACK-UP FRVOL=EMP004 TOVOL=HSM004 TRACKS=
        52 RC= 19, REASON= 12, AGE= 1, DSN=
```

The following message, and other instances of ARC0734I with non-zero return codes, indicates that the operation failed for a data set. That data set was not backed up or migrated; it may be uncataloged, cataloged but nonexistent, or ineligible for backup or migration.

```
ARC0734I ACTION=MIGRATE FRVOL=WORK02 TOVOL= *** TRACKS=
        0 RC= 20, REASON= 12, AGE=107, DSN=
```

Input Data Sets

Output management reads the DFHSM and DFDSS message logs. These logs must be written to DASD data sets by DFHSM and DFDSS.

Your DFHSM initialization parameters must include the ACTLOGTYPE(DASD) parameter so DFHSM will write the log records to DASD data sets used by the OPRTR000 program.

Note: DFHSM initialization parameters are usually stored in the ARCCMD $_{xx}$ member of your SYS1.PARMLIB data set.

If you use the alternative logging method with the ACTLOGTYPE(SYSOUT) parameter, your log data will not be available for the OPRTR000 program. See the *MAINVIEW SRM Reporting User Guide* for more information.

DFHSM uses the following naming convention for the log data sets:

`HSMACT.Hhostid.actLOG.Dyyddd.Thhmmss`

The variables in the naming convention are defined as follows:

- *hostid* is the one-character ID of the host machine
- *act* identifies the DFHSM activity as follows:
 - BAK backup
 - MIG migration
 - CMD command
 - DMP dump)
- *yyddd* is the Julian date when the data set was produced
- *hhmmss* is the time of day when the data set was produced

Note: You can set the high-level qualifier of the DFHSM activity log data sets in the HSMACTID parameter in the SMMSYS $_{xx}$ member of the SYS1.PARMLIB data set.

The DFHSM data sets are dynamically allocated by output management and do not require any entries in the execution JCL. You can use the DAYS/DATE and HOST parameters on output management control cards to specify which DFHSM logs should be used as input. DFHSM logs that have been migrated are recalled to ML0 for input processing.

Note: A maximum of 255 DFHSM logs can be input to any single execution of output management.

With DFDSS, the output data set for messages must be specified on the SYSPRINT DD statement of the DFDSS job. DFDSS does not require any naming standards. DFDSS data sets must be identified explicitly on the OPMIN DD statement.

Output management allows data to be extracted from selected DFHSM or DFDSS messages and substituted into *skeleton* statements. (This is conceptually similar to the ISPF JCL skeleton facility.) These skeleton statements, which are normally JCL or control card prototype statements, are read from partitioned data set (PDS) members. The input skeleton PDS is identified on the OPMSKL DD statement. The member name is specified on the SAVE or START output management control card. (See “Skeleton Processing” on page 6-28 for a complete description.)

Output Data Sets

Output management produces two types of output; reports and generated JCL and control cards.

Report Output

Output management generates two different report types that are identified by the REPORT and ERROR parameters in the output management specification requests. DFHSM and DFDSS messages can be filtered into either category; the distinction between normal reporting and error reporting depends on how you define it.

Reports are written to PDSs that output management dynamically creates. Each report PDS is given a name unique to the current date. Multiple executions of output management on the same day adds members to any existing reports PDS for that day. These output data sets use the following naming convention:

```
opmhlq.OUTPUT.DFHSM.Dyymmdd  
opmhlq.OUTPUT.DFDSS.Dyymmdd
```

The variables in the naming convention are defined as follows:

- *opmhlq* is the output management high-level qualifier specified in the OPMHLQ parameter of the SMMSYSxx member
- *yymmdd* is the current date

Report members in the output PDS use the following naming convention:

s h r hmmm

The variables in the naming convention are defined as follows:

- *s* identifies the source:
 - B DFHSM backup log
 - M DFHSM migration log
 - C DFHSM command log
 - D DFHSM dump log or DFDSS output
- *h* identifies the ID of the host machine
- *r* identifies the type of output
 - Report
 - Error
- *hmmm* identifies the time of day when output management produced the member

Example

M2R1503 migration report for host 2, run at 3:03 P.M.

BAE0811 backup error report for host A, run at 8:11 A.M.

System Parameters

Table 6-2 lists the parameters in the SMMSYS_{xx} PARMLIB member that are used to define and control output management.

Table 6-2 System Parameters

Parameter	Required	Description
OPMDIR= <i>nnn</i>	No	specifies the number of directory blocks allocated for the output data set
OPMHLQ= <i>xxxxxxxxxx</i>	No	specifies the output management high-level qualifier
OPMPRIM= <i>nnnn</i>	No	specifies the primary allocation amount of the output data set
OPMSEC= <i>nnnn</i>	No	specifies the secondary allocation amount of the output data set

OPMDIR=

Purpose: Specifies the primary directory blocks allocated for the output data set created by the output management facility.

Syntax: OPMDIR=*nnnn*
where *nnn* is 1–999

Required: No

Default: OPMDIR=10

OPMHLQ=

Purpose: Specifies the high-level qualifier for the data sets created by the output management facility.

Syntax: OPMHLQ=*xxxxxxxx*
where *xxxxxxxx* is any 1–8 character string.

Required: Only if the MAINVIEW SRM output management facility is to be used.

Default: None

OPMPRIM=

Purpose: Specifies the primary allocation amount of the output data set created by the output management facility.

Syntax: OPMPRIM=*nnnn*
where *nnnn* is 1–9999 tracks

Required: No

Default: OPMPRIM=10

OPMSEC=

Purpose: Specifies the secondary allocation amount of the output data set created by the output management facility.

Syntax: OPMSEC=*nnnn*
where *nnnn* is 1–9999 tracks

Required: No

Default: OPMSEC=20

JCL and Control Card Output

Output management allows data to be extracted from selected DFHSM and DFDSS messages and substituted into *skeleton* statements (normally JCL and control card images). The output that contains the skeleton text and substituted data is written to the sequential data set identified on the OPMSUB DD statement. The member name is specified on the SAVE control card. (See “Skeleton Processing” on page 6-28 for more information.)

You can also submit translated skeleton output directly to OS/390 for execution. An INTRDR DD statement is required in the JCL.

Execution JCL

Figure 6-2 shows the JCL that is required to execute output management. MAINVIEW SRM must be active to run output management.

Figure 6-2 Sample JCL for Output Management (member JCLOPM in the BBSAMP library)

```
//STEP10 EXEC PGM=OPRTR000,REGION=2M
//STEPLIB DD DSN=?$BBLINK,DISP=SHR <== MAINVIEW SRM LOADLIB
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSALLDA,SPACE=(CYL,(10,10))
//SYSUT2 DD UNIT=SYSALLDA,SPACE=(CYL,(10,10))
//OPMIN DD DSN=DFDSS.INPUT.FILE,DISP=SHR <== DFDSS MESSAGES
//OPMSKL DD DSN=?$OPMSKEL,DISP=SHR <== OPM SKELETONS
//OPMSUB DD DSN=?$OPMCARDS, <== OPM GENNED CARDS
// DISP=(NEW,CATLG),UNIT=SYSALLDA,
// SPACE=(TRK,(5,1))
//INTRDR DD SYSOUT=(A,INTRDR),DCB=BLKSIZE=80
//SYSIN DD *
/*
```

Table 6-3 describes the DD statements in Figure 6-2. (See also member JCLOPM in the BBSAMP library.)

Table 6-3 DD Statement Descriptions

DD Statement	Description
OPMIN	specifies the DFDSS message data set This DD statement is required if DFDSS is specified in any output management control statement. (DFHSM input data sets are allocated dynamically and do not require JCL specification.)
OPMSKL	specifies an input PDS containing skeleton specifications This DD statement is required when either SAVE or START is specified with an ACTION statement.
OPMSUB	specifies an output sequential data set for card images generated from skeleton statements This DD statement is required when SAVE is specified with an ACTION statement.
INTRDR	specifies an internal reader This DD statement is required when START is specified with an ACTION statement.
SYSIN	contains the control statements for output management processing

Control Statements

You define output management processing by specifying a series of control statements that contain a variable number of parameters. Like other MAINVIEW SRM functions, parameters are either action parameters or selection parameters (see Figure 6-3 on page 6-23).

You use *action* parameters to perform these tasks:

- specify the activity to be performed and the source of data

For example, BACKUP indicates that backup messages from DFHSM data set HSMACT.Hx.BACLOG.Dyyddd.Thhmmss are processed.

- specify the type of output to be produced

For example, REPORT indicates that a report of selected messages is produced.

- specify skeleton usage

For example, SAVE specifies the skeleton for building the output with data extracted from message input.

- specify the data to be extracted from messages and substituted into skeleton lines

For example, %VOL=AFTER("VOLUME=",6) indicates that a skeleton variable %VOL has the six-character string following "VOLUME=" in the selected messages.

You use *selection* parameters to perform these tasks:

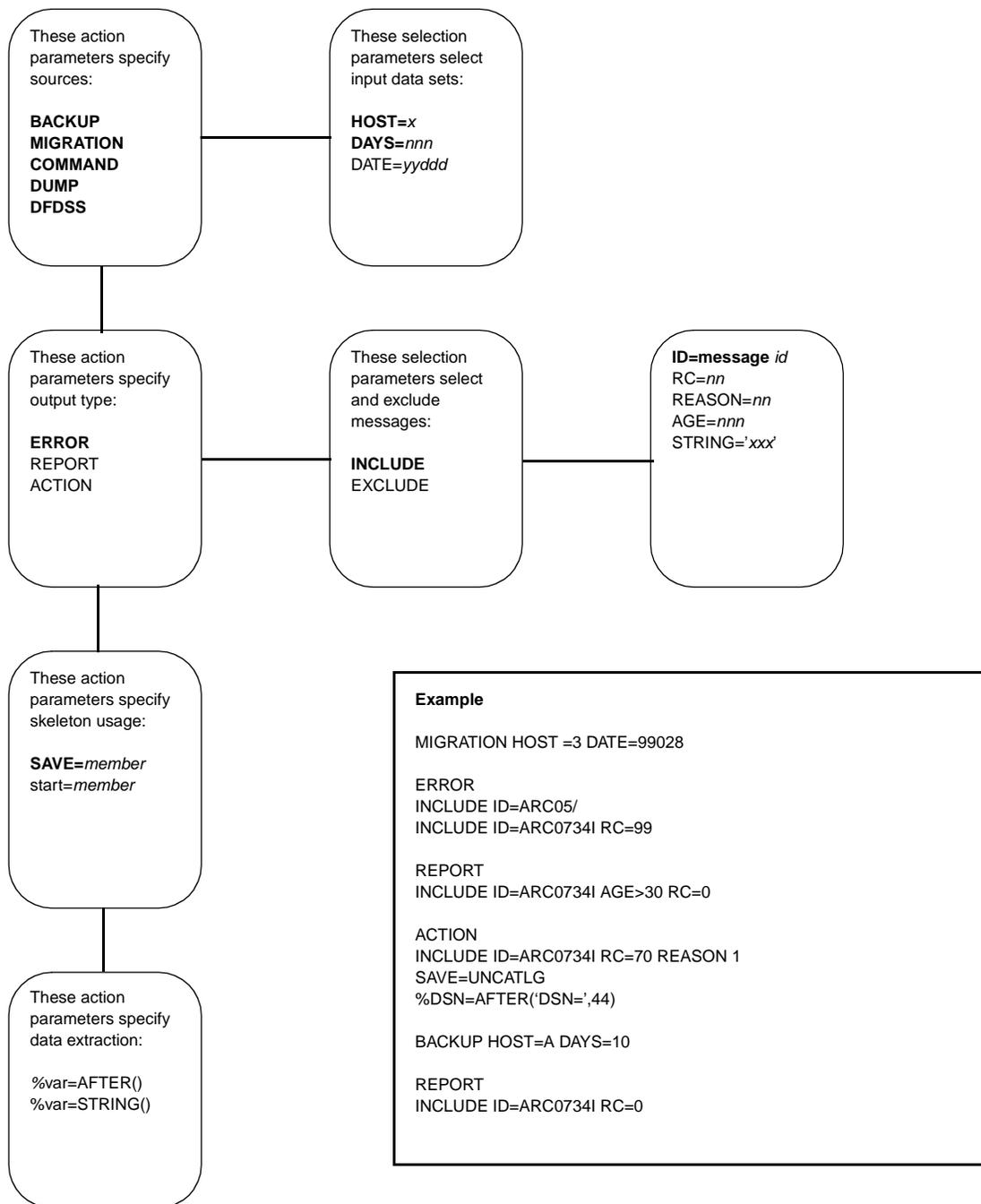
- specify the individual DFHSM logs to be read

For example, HOST=A DATE=98030 specifies that logs from host machine A created on 1/30/98 are to be processed.

- Specify messages to be included or excluded

For example, INCLUDE ID=ARC0706I indicates that all ARC0706I messages are included for processing.

Figure 6-3 DFHSM Output Management Action and Selection Parameters Overview



Syntax for an Output Set

A single execution of the output management program can produce multiple sets of output, and each output set can produce 1 to 3 reports. A single execution of output management can produce only one output set for a single host, but multiple hosts can be processed in each run.

The syntax for an output set consists of the following parameters:

- one action parameter (BACKUP, MIGRATION, DFDSS, and so on) that specifies input from DFHSM or DFDSS
- one or more action parameters (ERROR, REPORT, ACTION) that specifies output to reports or card image generation
- for each output action parameter, one or more selection parameters (INCLUDE, EXCLUDE, and so on) and one or more keywords (ID, RC, REASON, and so on)
- for each card image generation action parameter (SAVE, START), one action parameter specifying a skeleton member

Some parameters require a keyword followed by a value, while other parameters require a value or stand alone, as shown in the following examples:

Example

REPORT: no value follows the REPORT parameter

INCLUDE ID=ARC0523I: no value follows the INCLUDE parameter, but the ID keyword must be followed by a value

Observe the following additional conventions when using parameters:

- Parameters can begin in any column.
- A single parameter must be contained on one line.
- Multiple parameters can be on the same line, or separate lines, except for INCLUDE/EXCLUDE specifications, which must be contained on one line.
- There is no line continuation character.
- The maximum line length is 80.

- Blank lines are ignored.
- Comment lines must start with an asterisk in column 1. There is no facility for embedded comments (all comments must be on a separate line).
- Like MAINVIEW SRM function filter statements, selection parameters on the same line are joined with the AND operator; selection parameters on separate lines are joined with the OR operator.
- Selection keywords cannot be duplicated after a single INCLUDE/EXCLUDE parameter. For example, the statement INCLUDE AGE>30 AGE<60 is invalid; only the last AGE keyword is used.

You can use the following features with message selection keywords. (Message selection keywords are always specified after an INCLUDE or EXCLUDE parameter).

- You can use name masking in the ID message selection parameter. The following name masking characters are available:
 - % any single numeric character
 - ? any single alphanumeric character, except the period character (.)
 - / any character string, any length
- you can use the following comparison operators in the RC, REASON, and AGE message selection parameters:
 - = (equals): parameter must equal the specified value
 - / (not equals): parameter must not equal the specified value
 - < (less than): parameter must be less than the specified value
 - > (greater than): parameter must be greater than the specified value

Parameter Specifications

The action parameters in Table 6-4 on page 6-26 identify the general input source. Each of these action parameters begins a specification for one output set. Other action and selection parameters (noted below) define the input and output information.

Table 6-4 Action Parameters

Action Parameter	Description
BACKUP	DFHSM backup logs are processed
MIGRATION	DFHSM migration logs are processed
DUMP	DFHSM dump logs are processed
COMMAND	DFHSM command logs are processed
DFDSS	DFDSS message data sets are processed

DFHSM source parameters also require the selection parameters shown in Table 6-5 to identify the range of data sets to be read. These parameters are not available for DFDSS processing.

Table 6-5 Selection Parameters

Selection Parameter	Description
DATE= <i>yyddd</i>	specifies an exact date in Julian format (<i>yyddd</i>) All non-empty DFHSM logs for that day are processed. A specific date must be given; no masking characters are allowed. No comparison operators other than = (equals) are allowed. DATE or DAYS must be specified.
DAYS= <i>nnn</i>	specifies a range of days Up to 731 days can be specified. (However, no more than 255 input data sets are processed.) This value specifies the number of days (starting with the current day) for which log data sets are read; for example, DAYS=2 reads logs from today and yesterday. DAYS=0 is invalid. No masking characters are allowed. No comparison operators other than = (equals) are allowed. DATE or DAYS must be specified.
HOST= <i>x</i>	specifies the host machine ID Allowable characters are 0-9 and A-Z. No masking characters are allowed. No comparison operators other than = (equals) are allowed. This is a required parameter for DFHSM requests (MIGRATION, BACKUP, COMMAND, DUMP); it is not necessary for DFDSS requests.

Example

The following examples show valid parameter specifications:

BACKUP DATE=98028 HOST=2	Process DFHSM backup logs generated on 1/28/98 for host machine 2.
MIGRATION DAYS=10 HOST=A	Process DFHSM migration logs generated during the past 10 days for host machine A.
DFDSS	Process all DFDSS message output contained in the data set(s) on DD statement OPMIN.
COMMAND HOST=C DAYS=5	Process DFHSM command logs generated over the past 5 days for host machine C.

The parameters in Table 6-6 identify the type of output. At least one of these parameters must be specified for each output set request (BACKUP, MIGRATION, COMMAND, DUMP, DFDSS). No more than one of each output type (REPORT, ERROR, ACTION) can be specified for a single output set.

Table 6-6 Output Parameters

Output Parameter	Description
REPORT	specifies that selected messages are to be written to a report output member <i>shRhhmm in opmhlq.OUTPUT.DFxxx.Dyyddd</i>)
ERROR	specifies that selected messages are to be written to an error output member <i>(shEhhmm in opmhlq.OUTPUT.DFxxx.Dyyddd)</i> Note that the difference between REPORT and ERROR depends on your specification.
ACTION	specifies that selected messages are used as a source of data to be extracted and substituted into skeleton statements

Message Inclusion Parameters

The parameters in Table 6-7 specify criteria used to select or exclude messages from the DFHSM and DFDSS logs. The selection parameters must be preceded by the INCLUDE or EXCLUDE parameter. (As with the MAINVIEW SRM filter selection parameters, the three-character abbreviations INC and EXC can be used.) A selection parameter can be used only once in any single INCLUDE or EXCLUDE parameter. All parameters on the same line are joined with the AND operator. Parameters on separate lines are joined with the OR operator.

Table 6-7 Message Inclusion Parameters (Part 1 of 2)

Parameter	Description
ID=xxxxxxx	specifies a message ID (1-8 characters) Name masking can be used. This is a character comparison. The leading three characters must be specified in the mask; for example, ID=ARC0734/ instead of ID=734/
RC= <i>nn</i>	specifies a numeric return code Name masking cannot be used. Alternative comparison operators (/, <, >) can be used. This parameter is matched to the RC= value in messages. Note that not all DFHSM/DFDSS messages show return codes. If a message does not contain the string RC=, it is not selected by this parameter.
REASON= <i>nn</i>	specifies a numeric reason code Name masking cannot be used. Alternative comparison operators (/, <, >) can be used. This parameter is matched to the REASON= value in messages. Note that not all DFHSM/DFDSS messages show reason codes. If a message does not contain the string REASON=, it is not selected by this parameter.

Table 6-7 Message Inclusion Parameters (Part 2 of 2)

Parameter	Description
AGE=nnnn	specifies a numeric age Name masking cannot be used. Alternative comparison operators (/, <, >) can be used. This parameter is matched to the AGE= value in messages. Note that not all DFHSM/DFDSS messages show age. If a message does not contain the string AGE=, it is not selected by this parameter.
STRING='xxxxx xxx'	specifies a string to find in the message text The value must be in single quotes (apostrophes). Name masking cannot be used.

Example

The following examples show valid parameter specifications:

BACKUP DATE=98028 HOST=2 REPORT INC ID=ARC072/	Process DFHSM backup logs generated on 1/28/98 for host machine 2. Report all ARCO720-series messages.
MIGRATION DAYS=10 HOST=A ERROR INC RC/0 INC REASON>4	Process DFHSM migration logs generated during the past 10 days for host machine A. Report all messages with a non-zero return code or a reason code greater than 4.
MIGRATION DAYS=2 HOST=1 REPORT INC ID=ARC0522I INC ID=ARC0523I ERROR INC ID=ARC0734I RC>0	Process DFHSM migration logs generated today and yesterday on host machine 1 (the default). Report all volume start/stop messages (ARCO522-ARCO523). Report all data set migration messages (ARCO734) with a return code greater than zero.

Skeleton Processing

The ACTION parameter allows data to be extracted from selected messages and substituted into skeletons. This permits automatic creation of job streams to correct problems revealed by DFHSM or DFDSS processing. JCL and control cards can be created and saved in a PDS member for later use or submitted for immediate execution.

Using Variables in Skeletons

Skeleton members are read from the PDS that the OPMSKL DD statement identifies. Each skeleton member can have multiple lines; each line can have multiple *variables*. Each *variable* corresponds to an action parameter that specifies the data in the DFHSM or DFDSS message that will be extracted. The extracted data is substituted for the variable in the skeleton. For each message selected for processing the translated skeleton lines are added to the end of the output data set.

Variable names are four characters long; the first character must be a percent sign (%). Any variable in the skeleton that is not matched in the action parameters is written out unchanged (that is, the variable name is written). An unassigned variable (one that is specified in the action parameters but is not assigned a value) causes the skeleton line to be written with no variable substitution, even if other variables exist that have values.

Substitution of a variable's value in the skeleton statement causes all characters to the right of the variable to shift right. If any non-blank characters are shifted farther than column 72, the warning **** ERROR** is added to the translated line. Translated skeleton members are written to a sequential data set (DD OPMSUB) or immediately submitted for execution (DD INTRDR).

Use the action parameters in Table 6-8 to specify skeleton processing.

Table 6-8 Action Parameters Used in Skeleton Processing

Parameter	Description
SAVE	specifies that translated skeleton statements are saved in the sequential output data set (DD OPMSKL and OPMSUB must be specified in the JCL.) Only one SAVE or START can be specified in a single ACTION request.
START	specifies that translated skeleton statements are submitted for immediate execution (DD OPMSKL and INTRDR must be specified in the JCL.) Only one SAVE or START can be specified in a single ACTION request.
MEMBER=xxxxxxx	specifies the name of an input PDS member containing skeleton specifications (DD OPMSKL must be specified in the JCL.)
%var=AFTER('string',length)	Specifies that the variable named <i>var</i> receives the text following <i>string</i> , until a blank or comma is found, for a maximum of <i>length</i> .
%var=STRING(offset,length)	specifies that the variable named <i>var</i> receives the text beginning at <i>offset</i> in the message for <i>length</i> bytes The offset must be specified one greater than the actual offset (with the first character counted as 1).

Example

The following examples show valid parameter specifications:

```
BACKUP DAYS=5 HOST=A
ACTION
INC ID=ARC0734I RC=70
REASON=1
SAVE MEMBER=UNCATLG
%NAM=AFTER('DSN=',44)
```

Explanation: Process DFHSM backup logs generated over the last 5 days for host machine A. For all ARCO734 messages with return code 70 and reason code 1 (cataloged non-existent data set), generate JCL to uncatalog the data set. The data set name is extracted from the message by taking the characters following the DSN= *text*.

```
The skeleton member UNCATLG
contains:
//STEP10 EXEC PGM=IEFBR14
//DD1 DD DSN=%NAM,
// DISP=(OLD,DELETE,DELETE)
```

In this JCL, the value of the JCL DSN parameter is, after substitution, the data set name from the ARCO734 message.

Message ARC0734I

For easier reading, the frequently occurring DFHSM message ARC0734I is reformatted before being written to the output report.

The DFHSM format is as follows:

```
ARC0734I ACTION=MIGRATE FRVOL=WORK01 TOVOL=HSM002
TRACKS=2 RC=0, REASON=0, AGE=6,
DSN=HSMACT.H1.BAKLOG.D94049.T072111
```

The reformatted message in output management reports is:

```
ARC0734 IMIGRATE RC=0 RS=0 AGE=6
DSN=HSMACT.H1.BAKLOG.D94049.T072111
WORK01/HSM002 TRK=2
```

Data extraction from message ARC0734I is based on the DFHSM format.

Example

The following example shows the parameter specifications for creating a weekly report showing DFHSM normal and unsuccessful expired backup processing.

BACKUP HOST=1 DAYS=7	Read DFHSM backup logs for the last seven days.
REPORT INC ID=ARC0680I INC ID=ARC0681I INC ID=ARC0734I STRING='EXBACKV' RC=0	For a success report, extract start and stop messages and all data-set-level messages for expired backup processing that report successful completion.
ERROR INC ID=ARC0734I STRING='EXBACKV' RC>0	For an error report, select all data set level messages that report some type of error.

Example

The following example shows the parameter specifications for creating a daily report showing DFHSM unsuccessful backup processing at the data set level and volume level.

BACKUP DAYS=1 HOST=B	Read DFHSM backup logs for the current day for host machine B. (DAYS=1 assumes this job runs after all daily DFHSM processing is complete.)
**** SHOW PROBLEMS IN BACKUP **** REPORT EXC ID=ARC0734I INC ID=ARC/ STRING='TERMINATE' INC ID=ARC/ STRING='FAIL' INC ID=ARC/ STRING='ERROR' INC ID=ARC/ STRING='INOP' INC ID=ARC/ STRING='CANNOT' INC ID=ARC/ STRING='DISABLE' INC ID=ARC/ STRING='ABEND'	Select all detail messages (messages that have a message ID) that show problems, except data-set-level messages.
** SHOW DATA SET BACKUP PROBLEMS **** ERROR INC ID=ARC0734I RC>0 STRING='BACK-UP'	Select all data-set-level backup messages that have a non-zero return code.

Example

The following example shows the parameter specifications for listing all data sets over the last 60 days that DFHSM identified as unsupported for migration.

<pre>MIGRATION DAYS=60 HOST=D REPORT INC ID=ARC0734I RC=99</pre>	<p>Read DFHSM migration logs for the last 60 days for host machine D. Select all ARCO734 messages with a return code of 99.</p>
--	---

Example

The following example shows the parameter specifications for generating JCL to correct problems with the DFHSM control data set records.

<pre>parameters MIGRATION DAYS=2 HOST=4 ACTION INC ID=ARC0734I RC=2 REASON=8 INC ID=ARC0734I RC=6 REASON=0 INC ID=ARC0734I RC=13 REASON=8 INC ID=ARC0734I RC=20 REASON=2 INC ID=ARC0734I RC=20 REASON=12 INC ID=ARC0734I RC=20 REASON=16 SAVE MEMBER=OPMFI %DSN=AFTER('DSN=',44) skeleton OPMFI HSEND FIXCDS D %DSN DELETE</pre>	<p>Select migration ARCO734 messages with a return code/reason code that indicates a problem in the DFHSM control data set.</p> <p>Extract the data set name from the message for substitution into the skeleton JCL in member OPMFI.</p> <p>PDS member OPMFI (in the data set identified by DD OPMSKL) provides a skeleton control card. The %DSN variable is replaced by the data set name.</p>
--	---

Example

The following example shows the parameter specifications for generating JCL to uncatalog data sets that do not exist on the cataloged volume. The second part shows the parameters for generating and submitting JCL to catalog data sets that exist but are not cataloged.

First Execution of Output Management Parameters

<pre>MIGRATION DAYS=5 HOST=1 ACTION INC ID=ARC0734I RC=70 REASON=1 INC ID=ARC0734I RC=07 REASON=8 SAVE MEMBER=UNCATLG %NAM=AFTER('DSN=',44) SAVE MEMBER=UNCATLG %NAM=AFTER('DSN=',44)</pre>	<p>Select migration ARCO734 messages with a return code/reason code that indicates that the data set VTOC entry was not found.</p> <p>Extract the data set name from the message for substitution into the skeleton JCL in member UNCATLG.</p>
<pre>skeleton UNCATLG //STEP10 EXEC PGM=IEFBR14 //DD1 DD DSN=%NAM, // DISP=(OLD,UNCATLG,KEEP)</pre>	<p>PDS member UNCATLG (in the data set identified by DD OPMSKL) provides skeleton JCL. The %NAM variable will be replaced by the data set name.</p>

Second Execution of Output Management Parameters

<pre>MIGRATION DAYS=5 HOST=1 ACTION INC ID=ARC0734I RC=30 REASON=0 START MEMBER=CATLG %NAM=AFTER('DSN=',44) %VOL=AFTER('FRVOL=',6)</pre>	<p>Select migration ARCO734 messages with a return code of 30 and reason code of 0 (data set not cataloged).</p> <p>Extract the data set name and volume ID from the message for substitution into the skeleton JCL in member CATLG. The START keyword executes the tailored JCL.</p>
<pre>skeleton CATLG //CLEANUP JOB //STEP10 EXEC PGM=IEFBR14 //DD1DD DSN=%NAM,VOL=SER=%VOL, //DISP=(OLD,CATLG,KEEP),UNIT=SYSDA</pre>	<p>PDS member CATLG (in the data set identified by DD OPMSKL) provides skeleton JCL. The %NAM variable is replaced by the data set name; &VOL is replaced by the volume serial number.</p>

DFHSM Output Management Views

DFHSM and DFDSS messages can be written to data sets where you can view and edit them using the DFHSM Output Management views.

Follow these steps to access the Output Management views:

- Step 1** From the EZSRM Menu, select **DFSMS/HSM**.
- Step 2** Select **DFHSM Output Mgmt** to advance to the EZHSM Menu.

Table 6-9 describes Output Management views, including how to invoke them and which actions are available. For more information about the views, see the *MAINVIEW SRM Reporting User Guide*.

Table 6-9 DFHSM Output Management Views

View	Description	Invocation	Actions
Data Set List view	lists the data sets containing DFHSM and DFDSS messages Each data set listed in the DFHSM Output Management Data Set List view has one or more members associated with it. These members contain the messages from DFHSM or DFDSS that deal with certain specific areas, such as backup or migration.	HSMOMDS	/ Display members DEL Delete data set
Member List view	shows the source and type of messages contained in each member of the data set	HSMOMML DSN (of Output Management data set)	B Browse member E Edit member

Chapter 7 VTOC Scan Processing

This chapter presents the following topics:

How VTOC Scan Processing Works	7-2
SMVSCFxx Filter Member	7-3
SMMSYSxx member	7-4
VTOC Scan Extract File	7-8

How VTOC Scan Processing Works

The VTOC scan process provides you the ability to

- collect data in a master collection data set
- collect data in a user-specified collection data set
- filter data set collection
- save selection criteria
- apply saved selection criteria to another group of data sets
- execute in batch

The master collection data set is a continuously updated collection data set. When the MAINVIEW SRM Reporting component performs a collection against the master, it first determines the volumes that qualify for the collection. All data set records from the qualifying volumes are then dropped from the master. The volumes are then processed and new data set records are built and written to the master. This method of update allows you to split your collection into more manageable runs that are under your control.

The generated name of the master collection data set is the VSCAN_OINDX value, with .MASTER appended. The non-masters are the VSCAN_OINDX value, with .SMVSCF xx appended, plus date and time.

VTOC scan uses the master collection data set when you do not specify the collection data. You can choose a specific collection data set using the SETSRM action command with the complete data set name, the SMVSCF xx filter name suffix, or with a blank. A blank value uses the master collection data set. When you specify a filter name suffix, the Reporting component uses the most recent collection data set that was created from that filter member.

VTOC scan reports execute against the records in the master collection data set. To apply different selection criteria to the *master* collection data set, you can use the following:

- current VTOC Scan Data Set Statistics Entry view
- customized views
- SETSRM command

To apply the same selection criteria to *different* collection data sets, you can select a different collection data set from the VTOC Scan Collection Data Set View or use the SETSRM command.

Initiating a VTOC Scan

If you choose to manually initiate a VTOC scan, issue the following command:

```
/SVOSPASname ,SVOS VSCAN ,SUF=xx
```

The variable *SVOSPASname* is the product address space (PAS) name, and the variable *xx* is the suffix of the SMVSCF*xx* member to use in the scan.

Note: All modify commands used in MAINVIEW SRM require the following format:

The SVOS PAS name followed by a comma, the word SVOS followed by a space, and then the command.

The data set name that contains the scan output is indicated in a message appearing in the SVOS job log in response to this command.

SMVSCFxx Filter Member

Table 7-1 shows the filter keywords in the SMVSCF*xx* member.

Table 7-1 Filter Keywords in the SMVSCFxx Member (Part 1 of 2)

Parameter	Description
DSN_MASK=xxxxxxxxxx	specifies the data set name or mask
DSN_TYPE=x	specifies the data set type
END_UNIT=nnnn	specifies the ending unit address range
MASTER=YES/NO	specifies whether the collected VTOC scan output will go into a completely new collection data set or into the master collection data set
MNT_STATUS=xx	specifies the volume mount status
MRG_CATINFO=YES/NO	specifies whether to include catalog information in the collected statistics
MRG_SGCINFO=YES/NO	specifies whether to include application data in the collected statistics
POOL=	specifies the user pool mask used to filter volume and data set records processed by VTOC scan collection
RECORD_TYPE=x	specifies whether to generate data set or volume records
SGC_APPL=	specifies the application name mask used to filter volume and data set records processed by VTOC scan collection
SMS_GROUP=	specifies the SMS group name or mask
SMS_STATE=xxxxxxxxxx	specifies the volume's SMS status

Table 7-1 Filter Keywords in the SMVSCFxx Member (Part 2 of 2)

Parameter	Description
START_UNIT= <i>nnnn</i>	specifies the starting unit address range
SUBPOOL=	specifies the subpool mask used to filter volume and data set records processed by VTOC scan collection
VOLUME= <i>xxxxxxx</i>	specifies the volser or mask

The MASTER=YES/NO keyword in the SMVSCFxx member indicates to VTOC scan if the collected output will go into a completely new collection data set or into the master collection data set.

When you specify MASTER=YES, filtering is only valid at the volume level. All data sets within a volume must be collected. This means only the default settings can be specified on the DSN_MASK, DSN_TYPE, SGC_APPL, and RECORD_TYPE keywords. For each volume collected, the old volume data (volume and all its data set records) is replaced with the new data on the master.

When you specify MASTER=NO, all filter criteria is available for specification. The collected data is written out to a new collection data set.

SMMSYSxx member

Use the parameters listed in the following tables in the SMMSYSxx member to specify ranges of elements that are used to produce system-wide data set reports.

- Table 7-2 on page 5 lists age ranges
- Table 7-3 on page 7-6 lists size ranges
- Table 7-4 on page 7 lists percentage-used ranges

Table 7-2 lists the parameters that specify the age ranges used to produce the System-Wide Data Set Age reports.

Table 7-2 Age Range Parameters

Parameter	Description
VSCAN_AGER1	specifies the high end of age range one Data sets that have not been used between zero days and this value fall into AGER1. The value is required and is a 1–3 digit number. The number must be less than the remaining ranges.
VSCAN_AGER2	specifies the high end of age range two Data sets that have not been used between AGER1 and this value fall into AGER2. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER3	specifies the high end of age range three Data sets that have not been used between AGER2 and this value fall into AGER3. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER4	specifies the high end of age range four Data sets that have not been used between AGER3 and this value fall into AGER4. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER5	specifies the high end of age range five Data sets that have not been used between AGER4 and this value fall into AGER5. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER6	specifies the high end of age range six Data sets that have not been used between AGER5 and this value fall into AGER6. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER7	specifies the high end of age range seven Data sets that have not been used between AGER6 and this value fall into AGER7. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER8	specifies the high end of age range eight Data sets that have not been used between AGER7 and this value fall into AGER8. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. When set to zero, the range is ignored. However, all remaining ranges must also be set to zero.
VSCAN_AGER9	specifies the high end of age range nine Data sets that have not been used between AGER8 and this value fall into AGER9. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding ranges and less than the remaining ranges. Data sets that exceed range 9 fall into range 10, which is automatically set and cannot be set by a user.

Table 7-3 lists the parameters that specify the size ranges used to produce the System-Wide Data Set Size Range reports.

Table 7-3 Size Range Parameters

Parameter	Description
VSCAN_SIZR1	specifies the high end of size range one Data sets that are between zero and this value fall into SIZR1. The value is required and is a 1–9 number (in kilobytes). The value must be less than the remaining sizes.
VSCAN_SIZR2	specifies the high end of size range two Data sets that are between SIZR1 and this value fall into SIZR2. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR3	specifies the high end of size range three Data sets that are between SIZR2 and this value fall into SIZR3. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR4	specifies the high end of size range four Data sets that are between SIZR3 and this value fall into SIZR4. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR5	specifies the high end of size range five Data sets that are between SIZR5 and this value fall into SIZR6. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR6	specifies the high end of size range six Data sets that are between SIZR5 and this value fall into SIZR6. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR7	specifies the high end of size range seven Data sets that are between SIZR6 and this value fall into SIZR7. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR8	specifies the high end of size range eight Data sets that are between SIZR7 and this value fall into SIZR8. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero.
VSCAN_SIZR9	specifies the high end of size range nine Data sets that are between SIZR8 and this value fall into SIZR9. The value is optional and is a 0–9 digit number (in kilobytes). The number, if specified, must be higher than the preceding sizes and less than the remaining sizes. When set to zero, the size is ignored. However, all remaining sizes must also be set to zero. Data sets that exceed range 9 fall into range 10, which is automatically set and cannot be set by a user.

Table 7-4 lists the parameters that specify the percentage used ranges used to produce the System-Wide Data Set Percentage-Used Range reports. The report displays the percentage of a data set that is used (full). For VSAM data sets, the percentage used is calculated from the high-allocated RBA and high-used RBA values for each volume. These values are obtained from the catalog.

Table 7-4 Percentage-Used Parameters (Part 1 of 2)

Parameter	Description
VSCAN_PCTR1	specifies the high end of percentage used range one Data sets that are between zero and this value fall into PCTR1. The value is required and is a 1–3 digit number. The value must be less than the remaining percentage used values.
VSCAN_PCTR2	specifies the high end of percentage used range two Data sets that are between PCTR1 and this value fall into PCTR2. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR3	specifies the high end of percentage used range three Data sets that are between PCTR2 and this value fall into PCTR3. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR4	specifies the high end of percentage used range four Data sets that are between PCTR3 and this value fall into PCTR4. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR5	specifies the high end of percentage used range five Data sets that are between PCTR4 and this value fall into PCTR5. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR6	specifies the high end of percentage used range six Data sets that are between PCTR5 and this value fall into PCTR6. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR7	specifies the high end of percentage used range seven Data sets that are between PCTR6 and this value fall into PCTR7. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.

Table 7-4 Percentage-Used Parameters (Part 2 of 2)

Parameter	Description
VSCAN_PCTR8	specifies the high end of percentage used range eight Data sets that are between PCTR7 and this value fall into PCTR8. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero.
VSCAN_PCTR9	specifies the high end of percentage used range nine Data sets that are between PCTR8 and this value fall into PCTR9. The value is optional and is a 0–3 digit number. The number, if specified, must be higher than the preceding values and less than the remaining values. When set to zero, the values is ignored. However, all remaining values must also be set to zero. Data sets that exceed range 9 fall into range 10, which is automatically set and cannot be set by a user.

VTOC Scan Extract File

The VTOC Scan extract file provides you with a file layout against which you can run user-defined reports. The VTOC Scan extract file is created in a batch job. Use the following steps to create and use the VTOC Scan extract:

- Step 1** Execute the VTOC Scan (for instructions, see “Initiating a VTOC Scan” on page 7-3).
- Step 2** Modify and submit the VTOC Scan Extract File JCL in *?prefix.BBSAMP(SVWXTJCL)*.

```
//JOBNAME JOB (ACCOUNT)
//* * * * *
//*
//*          CREATE VTOCSCAN EXTRACT FILE
//*          -----
//*
//* REPLACE <LIBHLQ> WITH THE HLQ SELECTED FOR THE LIBRARIES
//*
//* REPLACE <USERGDG> WITH THE NAME OF A GENERATION DATA GROUP
//*
//COSSR400 EXEC  PGM=COSSR400
//STEPLIB DD     DISP=SHR,DSN=<LIBHLQ>.BBLINK
//VTOCDATA DD    DSN=<USERGDG>( +1 ),UNIT=SYSALLDA,
//              SPACE=(CYL,(100,50),RLSE),DISP=(,CATLG),
```

- Step 3** Use the resulting layout and your report writer software to create the desired report(s). The resulting layout should look similar the the one that follows.

Dec	Offsets	Length	Type	Description
	0	1	CHARACTER	RECORD ID 1 = VOL, 2 = DSN
----- Volume Record -----				
	1	6	CHARACTER	VOLUME
	7	1	CHARACTER	Alignment Filler
	8	4	FULLWORD	DATA SET RECORDS (TOTAL)
	12	4	FULLWORD	DATA SET RECORDS (NOT COMPLETE)
	16	3	CHARACTER	MOUNT STATUS
	19	2	HEXADECIMAL	UCB
	21	8	CHARACTER	DEVICE TYPE
	29	6	CHARACTER	EMULATED TYPE
	35	4	HEXADECIMAL	PERCENTAGE FULL
	39	1	CHARACTER	Alignment Filler
	40	4	FULLWORD	LARGEST PRIMARY ALLOCATION (CYL)
	44	4	FULLWORD	LARGEST PRIMARY ALLOCATION (TRK)
	48	4	FULLWORD	PERCENTAGE VTOC FULL
	52	4	FULLWORD	FREE DSCB(S) COUNT
	56	4	FULLWORD	FREE CYLINDERS COUNT
	60	4	FULLWORD	FREE TRACKS COUNT
	64	4	FULLWORD	FREE VIR COUNT
	68	4	FULLWORD	LARGEST CONTIGUOUS CYLS FREE
	72	4	FULLWORD	LARGEST CONTIGUOUS TRKS FREE
	76	4	FULLWORD	VTOC SIZE IN TRACKS
	80	4	FULLWORD	FRAGMENTATION INDEX
	84	30	CHARACTER	SMS GROUP
	114	3	CHARACTER	VTOC INDEX INDICATOR
	117	2	CHARACTER	SMS INDICATOR
	119	1	CHARACTER	PAGING DEVICE INDICATOR
	120	1	CHARACTER	VIO ELIGIBLE INDICATOR
	121	1	CHARACTER	CACHE WRITE FAST SUPPORT IND
	122	1	CHARACTER	CACHE SUPPORT IND
	123	1	CHARACTER	DUAL COPY SUPPORT IND
	124	1	CHARACTER	DASD WRITE FAST SUPPORT IND
	125	1	CHARACTER	CACHE WRITE FAST ACTIVE IND
	126	1	CHARACTER	CACHE ACTIVE ACTIVE IND
	127	1	CHARACTER	DUAL COPY ACTIVE INDICATOR
	128	1	CHARACTER	DASD WRITE FAST ACTIVE IND
	129	11	CHARACTER	PHYSICAL DISK ID
	140	4	CHARACTER	SUBSYSTEM ID
	144	12	CHARACTER	CONTROL UNIT SERIAL NUMBER
	156	8	CHARACTER	CONTROL UNIT TYPE-MODEL
	164	3	CHARACTER	DEVICE VENDOR
	167	10	CHARACTER	RAID TYPE
	177	1	CHARACTER	PAV INDICATOR

Dec	Offsets	Length	Type	Description
	178	2	HEXADECIMAL	ALIAS UCB COUNT
	180	4	HEXADECIMAL	RETURN CODE
	184	4	HEXADECIMAL	REASON CODE
	188	4	HEXADECIMAL	SUB-FUNCTION RETURN CODE
	192	4	HEXADECIMAL	SUB-FUNCTION REASON CODE
	196	4	HEXADECIMAL	SUB-FUNCTION FEEDBACK
	200	20	CHARACTER	ERROR MESSAGE
	220	4	FULLWORD	Volume Size in tracks
	224	4	FULLWORD	VIR size in tracks
	228	4	FULLWORD	PERCENTAGE VIR FULL
	232	4	FULLWORD	VVDS size in tracks
	236	4	FULLWORD	PERCENTAGE VVDS FULL
	240	4	FULLWORD	FREE EXTENT COUNT
	244	8	CHARACTER	Time Volume Scanned
	252	10	CHARACTER	Date Volume Scanned
	262	2	CHARACTER	Alignment Filler
	264	4	FULLWORD	POOL COUNT
	268	8	CHARACTER	POOL1 NAME
	276	1	CHARACTER	POOL1 TYPE
	277	8	CHARACTER	POOL2 NAME
	285	1	CHARACTER	POOL2 TYPE
	286	8	CHARACTER	POOL3 NAME
	294	1	CHARACTER	POOL3 TYPE
	295	8	CHARACTER	POOL4 NAME
	303	1	CHARACTER	POOL4 TYPE
	304	8	CHARACTER	POOL5 NAME
	312	1	CHARACTER	POOL5 TYPE
	313	8	CHARACTER	POOL6 NAME
	321	1	CHARACTER	POOL6 TYPE
	322	8	CHARACTER	POOL7 NAME
	330	1	CHARACTER	POOL7 TYPE
	331	8	CHARACTER	POOL8 NAME
	339	1	CHARACTER	POOL8 TYPE

Dataset Record

	1	44	CHARACTER	DATA SET NAME
	45	6	CHARACTER	VOLUME
	51	1	CHARACTER	Alignment Filler
	52	2	HALFWORD	VOLUME SEQUENCE NUMBER
	54	1	CHARACTER	CATALOGED INDICATOR
	55	1	CHARACTER	Alignment Filler
	56	2	HALFWORD	UCB
	58	2	CHARACTER	Alignment Filler
	60	4	FULLWORD	RECORD LENGTH
	64	4	FULLWORD	BLOCK SIZE
	68	4	FULLWORD	BLOCKING EFFICIENCY
	72	4	FULLWORD	BLOCKS PER TRACK

Dec	Offsets	Length	Type	Description
	76	4	FULLWORD	EXTENTS
	80	4	FULLWORD	ALLOCATION SIZE (IN K)
	84	4	FULLWORD	TRACKS ALLOCATED
	88	4	FULLWORD	TRACKS USED
	92	4	FULLWORD	TRACKS FREE
	96	4	FULLWORD	ALLOCATION PERCENTAGE USED
	100	4	FULLWORD	CA SPLITS
	104	4	FULLWORD	CI SPLITS
	108	4	FULLWORD	DAYS SINCE REFERENCED
	112	10	CHARACTER	LAST REFERENCE DATE
	122	10	CHARACTER	CREATION DATE
	132	10	CHARACTER	EXPIRATION DATE
	142	5	CHARACTER	RECORD FORMAT
	147	4	CHARACTER	DATA SET ORGANIZATION
	151	1	CHARACTER	CHANGED INDICATOR
	152	1	CHARACTER	REBLOCKABLE INDICATOR
	153	2	CHARACTER	SMS INDICATOR
	155	8	CHARACTER	SMS STORAGE CLASS
	163	8	CHARACTER	SMS DATA CLASS
	171	8	CHARACTER	SMS MANAGEMENT CLASS
	179	16	CHARACTER	SG-CNTL ACCOUNT LEVEL 1
	195	16	CHARACTER	SG-CNTL ACCOUNT LEVEL 2
	211	16	CHARACTER	SG-CNTL ACCOUNT LEVEL 3
	227	16	CHARACTER	SG-CNTL ACCOUNT LEVEL 4
	243	1	CHARACTER	Alignment Filler
	244	4	FULLWORD	RETURN CODE
	248	4	FULLWORD	REASON CODE
	252	4	FULLWORD	SUB-FUNCTION RETURN CODE
	256	4	FULLWORD	SUB-FUNCTION REASON CODE
	260	4	FULLWORD	SUB-FUNCTION FEEDBACK
	264	20	CHARACTER	ERROR MESSAGE
	284	30	CHARACTER	SMS Storage Group
	314	2	CHARACTER	Alignment Filler
	316	4	FULLWORD	Device Occupancy Percentage Used
	320	8	CHARACTER	Pool Name
	328	1	CHARACTER	Pool Type
	329	8	CHARACTER	DSN HLQ
	337	1	CHARACTER	Alignment Filler
	338	2	HALFWORD	Days Range 0 thru 10
	340	4	FULLWORD	Days Range specified in parmlib
	344	4	FULLWORD	Size Range specified in parmlib
	348	2	HALFWORD	Size Range 0 thru 10
	350	2	HALFWORD	Percent Used Range 0 thru 10
	352	4	FULLWORD	% Used Range specified in parmlib
	356	144	CHARACTER	Reserved for future growth

Chapter 8 Tape Reporting

The chapter presents the following topics:

Overview	8-2
Supported Environments	8-5
Tape Reports	8-5
Initiating an SVOS TSCAN	8-5
System Parameters	8-6

Overview

The tape reporting facility derives and consolidates information from your existing tape management software and other sources. Use the tape reports to accomplish the following tasks:

- determine if tape resources are being efficiently used
- manage and report on usage of the automated tape library (ATL)
- view both the performance and the utilization of the virtual tape system and take the necessary steps to improve both
- using capacity analysis, achieve workload balancing, thereby getting more benefit from fewer resources
- using global auditing, reduce tape-related errors and increases resource availability

Tape Reporting Features

The tape reporting facility reports on both physical and virtual tape library systems. It supports both the IBM and StorageTek virtual library systems and interfaces with CA-1, DFSMSrmm, and CONTROL-T tape management systems.

The tape reporting facility provides the following features:

- Auditing enables you to detect any “out of sync” conditions that exist between the various data sources; for example, the tape catalog, the OS/390 catalog, and so on.
- Query enables you to view information from the TSCAN results of your choice for a single entity, such as a tape volume, a tape unit, or a data set name.
- Reporting enables you to select an online view of the report in which you are interested and enables you to print the report using the MAINVIEW SRM Batch Reports process.

When you submit the SVOS TSCAN command, the tape reporting facility examines the status of the tape environment and extracts data from the available sources. This process extracts data and stores a refreshed set of tape-related information. This stored data (MVS data spaces and data sets) is available for faster scrutiny, such as further analysis, query, and reporting, from online or batch requests.

The tape reporting facility gathers information multiple sources for both real and virtual tape volumes (ATLs and conventional tape devices). Table 8-1 lists the software that the tape reporting facility interrogates when it gathers data that represents the tape environment.

Table 8-1 Interrogated Software

Software	Acronym Description	Vendor
AMS	Access Method Services	IBM
CA-1	Computer Associates Tape Management System	CA
CONTROL-T	CONTROL-T Tape Management System	BMC
DFSMSdfp	Storage Management Subsystem data facility product	IBM
DFSMShsm	Storage Management Subsystem hierarchical storage management	IBM
DFSMSrmm	Storage Management Subsystem removable media manager	IBM
HSC	Host Software Component	StorageTek
OAM	Object Access Method	IBM
SMF	System Management Facility	IBM
VTCS	Virtual Tape Control System	StorageTek

Some of the available data, such as AMS data set catalog entries and SMF record types 21, 73, 74, 94, and the STK user type are common to most environments. Some data will vary for different environments depending on the hardware and software that is used (for example, HSC and SMF user records or AMS library and volume catalog entries and SMF type 94 records). This alternate software usage can be separated into the following arbitrary categories. Table 8-2 on page 8-4 shows software by usage category.

- tape management
- storage management
- data archiving
- ATL real volumes
- ATL virtual tape volumes

Table 8-2 Software Usage Categories

Tape Management	Storage Management	Data Archiving	ATL (real)	ATL (virtual)
CA-1	DFSMSdfp	DFHSMShsm	HSC	VTCS and HSC
CONTROL-T	n/a	n/a	OAM	OAM
DFSMSrmm	device group,...	n/a	OAM	OAM

After the tape reporting facility extracts and merges the volume data, it produces a detailed tape inventory report for all real and virtual volumes. The report notes exceptions between data sources for common elements. For nonscratch volumes, it shows the data sets on the volume and the location of the volume. For StorageTek ATLs, the inventory lists such items as the library location, selection count, insertion time, and last selection and mount times.

Because the tape reporting facility records statistics at the device level, utilization and performance data is available and summarized at the ATL-, device-, and volume-level, and the data is grouped by user-defined applications.

The information on the reports enables you to locate performance problems caused by excessive or under-utilization of the storage media (tape, DASD, cache storage) or hardware (ATL, CU, device). You can use the information to make decisions about device selection and allocation, such as Automatic Class Selection (ACS) for management, storage, and data, to achieve these performance improvements:

- balanced work load
- reduced physical volume movement
- reduced virtual volume thrashing

You can also use the information to accomplish these tasks:

- make data-migration decisions for migration levels (MLs) that minimize delays that are caused by volume mounts that are associated with data set recall
- generate reports that provide the data you need to better plan and forecast tape resource needs
- redistribute shared tape devices to suit the work load
- identify a resource shortage (scratch volumes, empty cells, available cache, devices) before the shortage becomes critical

Supported Environments

The tape reporting facility provides information on most enterprise tape environments by accessing data using an API, data sets, or extract files.

For the IBM Automated Tape Library (ATL) and the IBM Virtual Tape Server (VTS), these data sources are not available. Therefore, the tape reporting facility accesses data from the tape catalog database (TCDB) and SMF 94 records. Updates to the SMF 94 records for these devices occurs hourly; therefore, the tape reporting facility reports on these devices on an hourly basis.

Tape Reports

The tape reporting facility generates a comprehensive set of reports, both scheduled in batch and ad hoc, which you can view online or print. Reports provide you with a complete picture of the current and anticipated state of all components of the tape system and provide data that is useful for accounting, capacity planning, auditing (error or exception reporting), performance tuning and troubleshooting. The tape reporting facility assists you in resolving problems related to the interaction of the various tape system components (TMC, VTS, and MVS) and in using the tape system to maximum efficiency.

Initiating an SVOS TSCAN

The TSCAN command initiates a tape scan collection. Based on your input, the system scans the tape catalog data set and extracts information about tape volumes and tape data sets that exist in the system. Information is also extracted from the tape silo, the VTS, and the HSM OCDS. Use the following command to begin a scan:

```
/SVOS TSCAN
```

To stop a TSCAN before completion, issue the following command:

```
ENDTSCAN
```

This data is built in linear data sets that have the date and time as a part of the name. These linear data sets are then processed to produce the TSCANLS view.

System Parameters

Table 8-3 lists the system parameters that you use to set up the tape reporting facility. For information about defining system parameters, see the *MAINVIEW SRM User Guide and Reference*.

Table 8-3 System Parameters

Parameter	Purpose
TAPE_CA1DSN=xxxxxxxxxxx	specifies the data set name of the CA1 data set
TAPE_CAT=(xxxxxxxx,xxxxxxxx,...)	specifies the tape management system(s) available for report generation The values are one or more of CONTROLT, CA1, or RMM. That is, TAPE_CAT=CONTROLT indicates that CONTROL-T is the only system that you are interested in. On the other hand, TAPE_CAT=(CONTROLT,RMM,CA1) indicates that you have all three systems and you want reports on all three.
TAPE_CCTLTH=xxxxxxxxxxx	specifies the high-level qualifier for the CONTROL-T data sets
TAPE_CHLQ=xxxxxxx	specifies the high-level qualifier for the TSCAN data sets; limited to 20 characters RELEASE <i>cannot</i> be turned on for TSCAN data sets.
TAPE_CPRI=nnnn	specifies the number of cylinders for the primary allocation
TAPE_CSEC=nnnn	specifies the number of cylinders for the secondary allocation
TAPE_CTLTRL=x	specifies the release number for Control T
TAPE_CVOL=(xxxxxx,xxxxxx,...)	specifies the volume serial number(s) of the volumes to used for the linear data sets, with a maximum of six volsers
TAPE_HSCCDS=xxxxxxxxxxx	specifies the name of the STK CDS data set
TAPE_RMMDSN=xxxxxxxxxxx	specifies the data set name for the RMM control data set
TAPE_SMFDS n =xxxxxxxxxxx	specifies 1-9 SMF data set names You can specify up to 9 SMF data sets. If you skip one, for example, specifying SMFDS4 and SMFDS6 but not SMFDS5, processing will stop at the first gap, which means that SMFDS6 would not be allocated. The SVOS TSCAN processes files created by IFSMFDP, not live VSAM files. If you do not want SMF data, do not specify the data set name.

Appendix A System Exits

This appendix presents the following topics:

Overview	A-2
Installing Exits	A-2
Exit Flow Example	A-3
Security Exit	A-4
Application Code Build Exit.	A-7
Add Exit	A-9
Check Exit.	A-10
Default Exit.	A-12
Select Exit.	A-13
Table of Sample Exits	A-15
Application Records	A-16
Global Work Area	A-18
Special Considerations for Exits.	A-18
DFHSM Exit Routines	A-19

Overview

The Application component of MAINVIEW SRM provides exits that allow you to extend the functions of the product. With the possible exception of the Security Exit, many installations can use the product without the use of exits. Table A-1 describes some of the available application system exits.

Table A-1 System Exits

Exit	Description
Application Code Build	Called each time an application code is built; this exit allows your installation to assign application codes programmatically. Note: In most cases, FLST and RLST parameters can provide the necessary capabilities and are recommended over the use of the Application Code Build Exit.
Add	Allows your installation to control updates to the current allocation amounts for an application.
Check	Allows your installation to check space allocation requests and budgets for an application.
Default	Invoked whenever an application is dynamically created; this exit allows your installation to override the defaults that are assigned to the application.
Security	Called when the BUDGET or BUDDSN TSO commands are invoked; this exit allows your installation to restrict the use of functions within the TSO command.
Select	Called once during the preprocessing exit and once during the postprocessing exit; this exit allows your installation to restrict application control to certain data sets or volumes. Note: FLST and RLST parameters provide the same capability; BMC Software recommends that you use them instead of the Select Exit.

Installing Exits

You must link each exit to a separate load module in an authorized library that can be accessed by SVOS started task and by any function that requires access to any of the application collector functions.

Note: Review “Special Considerations for Exits” on page A-18 before designating exits.

The following global parameters establish the link to the load module:

- SGC_ADDEXIT—load module name of the Add Exit.
- SGC_CHKEXIT—load module name of the Check Exit. If blank, all DASD allocations are monitored.

- **SGC_DEFEXIT**—load module name of the Default Exit. If blank, all applications created dynamically use the defaults from the CONTROL budget application.
- **SGC_KEYEXIT**—load module name of the Application Code Build Exit. If blank, all application codes are built from the fields specified when the application database was initialized. Application codes also can be constructed using the NEWAPPL statement in the selection language.
- **SGC_SECEXIT**—load module name of the Security Exit. If blank, application security is not in effect.
- **SGC_SELEXIT**—load module name of the Select Exit. If blank, space allocation requests are monitored. FLST and RLST parameters provide the same capability; BMC Software recommends that you use them instead of the exit.

Exit Flow Example

Exits can be called several times for each space allocation request. The best way to describe the exit flow is to consider the worst case. For this example, assume that a user is about to allocate a new data set. Allocation of this data set will cause new applications to be added to the application database.

During CHECK processing, the space allocation request performs these steps:

- calls the Select Exit, which is used to determine if a space allocation request should be tracked
- calls the Application Code Build Exit twice for each application code

The default application code construction rules are passed to the exit the first time. You can change the rules or bypass tracking this application code. The application code is passed to the exit the second time it is called. You can change the application code or bypass processing for this application.

- calls the Default Exit twice each time a new application is created

The first call passes the application code of the default application. You can change the application code of the default application. The second call is made after the new application is built. You can change any field in the application record. On either call, you can choose not to track the space allocation request for this application code.

- calls the Check Exit for each application with the amount of space requested and the application record

You can change the amount of space to be checked, skip the check, ignore the result, or reject the allocation.

During ADD processing, the space allocation request performs these steps:

- calls the Select Exit
- calls the Application Code Build Exit

This call must generate the same application codes that the preprocessing call generated.

- calls the Add Exit for each application with the amount of space requested and the application record.

You can change the amount of space to be updated or skip the updating of the application.

Security Exit

The Security Exit is called each time the BUDGET or BUDDSN TSO command is executed. Misuse of the TSO BUDGET command could compromise the integrity of the application database. For example, the BUDGET command can be used to list and change budget and allocation amounts. Since it would be impossible to control DASD space if all users could change application budgets at will, you should either use the exit or restrict access to the BUDGET command.

A mode flag that indicates whether the user wants to read, change, or add a record is passed to the Security Exit. The exit is called once with a special mode flag if the user wants to process the entire database. Both the original and updated copies of the records are presented to the exit. This allows the exit to identify any fields that have changed. The following are some uses for the Security Exit:

- to control privileges to add or update the application database
- to control privileges to reset the high-water marks
- to distribute database control
- to force the use of a backup copy of the database for reporting and query purposes

Controlling Access to Applications

The primary function of the Security Exit is to control access to the applications. The simplest way to accomplish this goal is to hard-code lists of user IDs that are allowed to read, change, or add selected applications. This approach can be difficult to maintain and does not allow for the distribution of authority.

If you have RACF, you can set up a new RACF class that has the application as the entry and have RACF maintain the access list. All RACF facilities are then available to maintain the list. The Security Exit simply issues a RACHECK. RACF access of READ, UPDATE, and ALTER can correspond to READ, CHANGE, and ADD in the system. RACF entry must be added and an access list that is associated with the RACF entry any time a new application is added.

Another technique is to encode or store in a user field the attributes of individuals who have permission to perform certain functions against the application. For example, a user could be tied to a group, which gives the user the ability to list all applications within that group.

You can also use the Security Exit to interface with other security systems such as CA-ACF2 and CA-TOP SECRET.

Controlling Application Data

An installation may need to restrict the ability to change fields in an application. For example, it may be advisable to restrict the ability to change any fields that may be used for capacity planning and costing.

Controlling Performance

An installation may need to restrict the use of commands that force the database to be read sequentially. For many processes, reading a backup copy of the database is acceptable, especially if the user is processing a large group of records. The user can allocate a database file with ISPF panels or with the TSO ALLOC command. In this case, the BUDGET command uses this locally allocated database instead of the active database.

This feature can be useful when the MAINVIEW SRM Reporting component is not active. When a user allocates an application database file, the high-order bit of BCVTFLG1 is set on. If a sequential read is required, the installation can force all users to use a backup database. Note that the system or DASD administrators should be allowed to process mass changes against the active database.

Register Conventions

On entry to the Security Exit, the registers contain the information shown in Table A-2. See Table A-8 on page A-15 for a list of the sample Security Exits on the product distribution tape.

Table A-2 Security Exit Register Contents

Register	Contents
0	<ul style="list-style-type: none"> • 0=Read • 4=Change • 8=Add • 20=Sequential Read
1	address of the updated or new application record; zero indicates a sequential read.
2	address of a 300-byte work area common to all exit routines; the high-order byte of the register contains x'01'.
3	address of the original application record; zero indicates add or sequential read.
10	address of the static area
11	address of global work area that is mapped by DSECT BUDWORK
13	address of a 72-byte save area; standard linkage conventions must be used.
14	return address
15	entry-point address of the exit

The Security Exit is returned upon completion with Register 15 containing one of the following codes:

- 0=allow the request to be completed
- 8=fail the request for this application

Application Code Build Exit

The Application Code Build Exit is called twice for each level that is associated with a data set. The default application-code-construction rules are passed to the exit on the first call. The application code is passed to the exit on the second call. The Application Code Build Exit can modify the default application-code-construction rules or replace the application code.

Note: BMC Software recommends that you use FLST and RLST parameters, which provide the MAINVIEW SRM Reporting component with a similar function.

FLST and RLST parameters and the Application Code Build Exit can be used to bypass processing on an application-code basis. FLST and RLST parameters or the Select Exit can be used to skip processing on a data set basis. Application codes can be constructed during application processing as follows:

- Default application codes are constructed using the default application-code-construction rules. FLST and RLST parameters can then be used to replace the default application code.
- The Application Code Build Exit can be called to make the final changes.

The following are some uses for the Application Code Build Exit:

- to extend the application code building facilities
- to include installation-unique information in application codes, such as special group codes that are derived using an algorithm
- to provide different application code logic for different groups of data sets
- to limit building an application to certain groups of data sets

Register Conventions

On the first entry to the Application Code Build Exit, the registers contain the information shown in Table A-3. Table A-8 on page A-15 contains a list of the sample Application Code Build Exits that are provided on the distribution tape.

Table A-3 Application Code Build Exit Register Contents

Register	Contents
0	zero
1	address of the default application-code-construction rules that are mapped by the DSECT KBF.
2	address of a 450-byte work area that is common to all exit routines; the high-order byte of the register contains x'03'.
10	address of the static area
11	address of the global work area that is mapped by the DSECT BUDWORK
13	address of a 72-byte save area; standard linkage must be used.
14	return address
15	entry-point address of the exit

On the second entry to the Application Code Build Exit, the registers contain the same information with the following exceptions:

- Register 0 contains 4.
- Register 1 contains the address of the application code.

When the Application Code Build Exit returns after processing is complete, register 15 contains one of the following codes:

- 0 use the application code that register 1 points to
- 4 skip this application code
- 8 skip the space allocation request

Note: If Register 0 contains zero (first call), Register 1 points to application-code flag fields. If Register 0 contains 4 (second call), Register 1 points to an application code.

Add Exit

The Add Exit is called during ADD processing. The exit is called once for each application code that is associated with the space allocation request. The amount of requested space is passed to the Add Exit. The amount is negative in the case of a SCRATCH or RENAME. The application is also passed to the exit. The installation can change the amount of space being requested or skip the update. The following are some uses for the Add Exit:

- to skip updating if the data set meets some installation standard
- to add a surcharge if the allocation is on certain volumes

Any changes made in the Check Exit should be replicated in the Add Exit; consider using the same code for both exits. The space allocation request cannot be rejected in the Add Exit because the space has already been allocated or deallocated.

Register Conventions

On the first entry to the Add Exit, the registers contain the information shown in Table A-4. Table A-8 on page A-15 contains a list of the sample Add Exits that are provided on the distribution tape.

Table A-4 Add Exit Register Contents

Register	Contents
0	address of the application record
1	amount of space requested (positive or negative) in kilobytes
2	address of a 450-byte work area that is common to all exit routines; the high-order byte of the register contains x'06'.
10	address of the static area; used by the SWA mod routine.
11	address of the global work area that is mapped by the DSECT BUDWORK
13	address of a 72-byte save area; standard linkage should be used.
14	return address
15	entry-point address of the exit

When processing is complete, the Add Exit returns with the following codes:

- Register 1—amount of space to be used for the update.
- Register 15:
 - 0 allow the update
 - 4 skip update processing for the application code
 - 8 skip the space allocation request

Check Exit

The Check Exit is called during CHECK processing. It can be used to check the current space allocation request, current allocation total, and budget for an application code. The exit is called once for each application code that is associated with the space allocation request. The amount of requested space is passed to the Check Exit. The application is also passed to the exit. The installation can change the requested space, ignore the result of the check, skip the check, or reject the space allocation request.

The following are some uses for the Check Exit:

- to skip checking if the data set meets some installation standard
- to add a surcharge if the allocation is on certain volumes
- to enforce date or time-of-day restrictions
- to enforce an installation standard on the maximum space any data set can request
- to reward users who allocate small data sets by excluding them from control under the application collector
- to impose a surcharge or discount if space is requested on certain volumes

You can implement these policies by using the Check and the Add Exits. Since the amount of space can be negative on a SCRATCH or RENAME, the exit should check the sign of the space amount before adding or subtracting any additional amounts.

- to enforce time-of-day or day-of-the week standards, because the Check Exit can reject a space allocation request

- to enforce installation standards that restrict the amount of space a data set can request

For example, it is not possible to limit with the application collector the total amount of space a data set may occupy eventually if all secondary requests are satisfied; the application collector can limit only the amount of requested space for this space allocation request.

Register Conventions

On entry to the Check Exit, the registers contain the information shown in Table A-5. Table A-8 on page A-15 contains a list of the sample Check Exits that are provided on the distribution tape.

Table A-5 Check Exit Register Contents

Register	Contents
0	address of the database Information (application) Record mapped by the DSECT DBI
1	amount of space requested (positive or negative) in kilobytes
2	address of a 450-byte work area that is common to all exit routines; the high-order byte of the register contains x'05'.
10	address of the static area
11	address of the global work area that is mapped by the DSECT BUDWORK
13	address of a 72-byte save area; standard linkage should be used.
14	return address
15	entry-point address of the exit

When processing is complete, the Check Exit returns with the following codes:

- Register 1—amount of space to be used for the check.
- Register 15:
 - 0 allow the check
 - 4 skip check processing for this application code
 - 8 skip the space allocation request
 - 12 reject the space allocation request

Note: This return causes the space allocation request to be rejected.

Default Exit

The Default Exit is used each time an application is created. The exit is called first to allow the installation to change the application code of the default (or model) application and a second time to alter specific fields within the new application. The CONTROL application is used as a model when there is no Default Exit. The maximum permanent space, temporary space, VSAM space, NAME field, and user fields are extracted from the model.

The Default Exit allows installations to override the application code of the model and to tailor fields in the application record. The following are some uses for the Default Exit:

- to determine the application code of the model based on user criteria
- to fill a user field with the TSO user ID of the creator
- to enforce installation standards that govern the maximum budget that any application can use

Budget Models

The Default Exit and the TSO BUDGET command can use models. The use of model applications enables installations to set up standards that can be used when creating budget applications. Application processing does not distinguish between real applications and model applications. It is up to the user to ensure that model applications are not used to control allocations. BMC Software suggests that the NAME or one of the user fields be set to a special code to ensure these applications are excluded from application total reports.

You can use the Default Exit to set up standard defaults for each user group. The Default Exit could then select the default application based on job accounting information, RACFGRP, or high-level qualifier.

Assuming an installation explicitly sets up all budget applications with the TSO BUDGET command, the budgets in the model application could be set to zero to reject any allocations performed against applications that were dynamically created. The Default Exit could be used to override this standard for selected situations.

The Default Exit also can be used to set the NAME and user fields in dynamically created applications.

Register Conventions

On the first entry to the Default Exit, the registers contain the information shown in Table A-6. Table A-8 on page A-15 contains a list of the sample Default Exits that are provided on the distribution tape.

Table A-6 Default Exit Register Contents

Register	Contents
0	address of the application code of the model application
1	address of the new application record
2	address of a 450-byte work area that is common to all exit routines; the high-order byte of the register contains x'02'.
10	address of the static area
11	address of the global work area that is mapped by the DSECT BUDWORK
13	address of a 72-byte save area; standard linkage should be used.
14	return address
15	entry-point address of the exit

On the second entry to the Default Exit, the registers contain the same information with the exception of Register 0, which contains zero.

When processing is complete, the Default Exit returns with one of the following codes contained in Register 15:

- 0 allow the application to be created
- 8 do not create an application. This application is ignored.

Select Exit

The Select Exit is called during CHECK and ADD processing. It is called once for each application code associated with a space allocation request. The exit can be used to decide if a request should be tracked by the application collector. FLST and RLST parameters provide the same capability; BMC Software recommends that you use them instead of the Select Exit. The Select Exit is used also by the BUDGET and BUDDSN TSO commands. The following are some uses for the Select Exit:

- restrict the application collector to certain data sets or volumes
- exclude certain user groups or programs
- phased implementation of application control

An installation may want to restrict the product to a set of data sets or volume serials or to exclude certain users from application control. The Select Exit or FLST and RLST parameters allow this type of control. All fields supported in the selection language are available in the Global Work Area. The address of the Global Work Area is passed to the exit.

Register Conventions

On the first entry to the Select Exit, the registers contain the information shown in Table A-7. Table A-8 on page A-15 contains a list of the sample Select Exits that are provided on the distribution tape.

Table A-7 Select Exit Register Contents

Register	Contents
0	zero
1	zero
2	address of a 450-byte work area that is common to all exit routines; the high-order byte of the register contains x'04'.
10	address of the static area
11	address of the global work area that is mapped by the DSECT BUDWORK
13	address of a 72-byte save area; standard linkage should be used.
14	return address
15	entry-point address of the exit

When processing is complete, the Select Exit returns with one of the following codes contained in Register 15:

- 0 allow the application collector to control this function
- 8 skip all processing

Table of Sample Exits

Table A-8 contains a list of sample exits delivered with the MAINVIEW SRM Reporting component. The example exits are not an exhaustive list of possible combinations of exit options, but rather serve as models that you can tailor to suit your environment.

Note: BMC Software encourages you to send application exits in machine readable format (3420 tapes, please). Exits that have universal application will be considered for inclusion in future releases.

Table A-1 describes sample exits that are located in the Installation Control library *?prefix.BBSAMP*.

Table A-8 **Sample Exits**

Exit name	Type	Description
BUDSEL01	SELECT	restricts application control to a particular job name
BUDSEL02	SELECT	restricts application control to a particular range of volumes; contains a table of volume serials (or partial volume serials) that are <i>not</i> to be tracked
BUDSEC01	SECURITY	issues an RACHECK to a RACF class of BUDGET by use the application code as the resource; allows an installation to use RACF to control the update of budget applications.
BUDSEC02	SECURITY	allows only the TSO user ID that matches the user ID stored in user field 1 to update the application. The exit also contains a list of user IDs that are not restricted. When the budget application is set up, the administrator sets user field 1 to the user ID of the TSO user who is allowed to update it. Since delegated users can change any field, they can update user field 1 when necessary.
BUDSEC03	SECURITY	prohibits users from changing the space allocation totals
BUDSEC04	SECURITY	issues a RACROUTE to a RACF class of BUDGET by using the application code as the resource. This function is identical to BUDSEC01, but it uses the RACROUTE macro instead of RACHECK.
BUDDEF01	DEFAULT	sets the application code of the default application to CONTROL1 for LEVEL 1, CONTROL2 for LEVEL 2, and so on; can be used if there are different defaults based on the LEVEL or application being created
BUDKEY01	APPLICATION CODE BUILD	builds an application code that consists of the address space ID; can be used to limit the total amount of temporary space a job can have during its execution. If there is more than one system, the SMF ID must be added to the application code.
BUDCHK01	CHECK/ADD	doubles the amount of space that is charged if the data set is allocated on certain volumes.; can be used to add a surcharge if the space is allocated on a volume with a cached controller or on a high speed channel The volume list is coded in the exit in the same manner as BUDSEL02.

Application Records

Table A-9 documents the format of the application record. An assembler DSECT for the application record is located in member DBI41 in the *?prefix.BBSAMP*.

Note: DPFP = floating point long format.

Table A-9 Application Record Format (Part 1 of 2)

Offset	Name	Type	LEN	Description
00 (00)	DBI	DSECT		
(00)	DBIKEY	CHAR	50	Application code
50 (32)	DBIFLAG1 DBIACT DBIDEL DBIMDEL DBIFMT41	HEX EQU EQU EQU EQU	1 X'80' X'40' X'20' X'08'	Status flag 1 Active Inactive Manually flagged for deletion DBI in Release 4.1.0 format
51 (33)	DBIFLAG DBIWARNM DBIMONM DBIREJM DBIDEF	HEX EQU EQU EQU EQU	1 X'80'X' 40' X'20'X' 00'	Status flag 2 Warning mode Monitor mode Reject mode Default mode
52 (34)	DBIRECID	HEX	4	Relative record number
	* * PERMANENT TOTALS *			
	DBIPERMS			
56 (38)	SPCMAX	DPFP	8	Budget amount — floating point
64 (40)	SPCCUR	DPFP	8	Current allocation — floating point
72 (48)	SPCHWM	DPFP	8	High-water mark —floating point
80 (50)	SPCFLAGS SPCPERM SPCNONE	HEX EQU EQU	1 X'80' X'40'	Space flags Add to perm Do not track
81 (51)		HEX	3	Slack
	* * TEMPORARY TOTALS *			
	DBITEMPS			
84 (54)	SPCMAX	DPFP	8	Budget amount— floating point
92 (5C)	SPCCUR	DPFP	8	Current allocation—floating point
100 (64)	SPCHWM	DPFP	8	High-water mark—floating point

Table A-9 Application Record Format (Part 2 of 2)

Offset	Name	Type	LEN	Description
108 (6C)	SPCFLAGS SPCPERM SPCNONE	HEX EQU EQU	1 X'80' X'40'	Space flags Add to perm Do not track
109 (6D)		HEX	3	Slack
	* * VSAM TOTALS *			
	DBIVSAMS			
112 (70)	SPCMAX	DPFP	8	Budget amount — floating point
120 (78)	SPCCUR	DPFP	8	Current allocation — floating point
128 (80)	SPCHWM	DPFP	8	High-water mark — floating point
136 (88)	SPCFLAGS SPCPERM SPCNONE	HEX EQU EQU	1 X'80' X'40'	Space flags Add to perm Do not track
137 (89)		HEX	3	Slack
	* * HSM TOTALS *			
	DBIHSMS			
140 (8C)	SPCFILL	HEX	8	Filler
148 (94)	SPCCUR	DPFP	8	Current allocation — floating point
156 (9C)	SPCHWM	DPFP	8	High-water mark — floating point
164 (A4)	SPCFLAGS SPCPERM SPCNONE	HEX EQU EQU	1 X'80' X'40'	Space flags Add to perm Do not track
165 (A5)		HEX	3	Slack
168 (A8)	DBIWARN	HEX	4	Warning percent
172 (AC)	DBINAME	CHAR	20	Application descriptive name
192 (C0)	DBIUSRF1	CHAR	8	User field 1
200 (C8)	DBIUSRF2	CHAR	10	User field 2
210 (D2)	DBIUSRF3	CHAR	10	User field 3
220 (DC)	DBICRTDT	HEX	4	Create date CCYYMMDD
224 (E0)	DBIDLUPD	HEX	4	Date last updated CCYYMMDD
228 (E4)		HEX	27	Slack

Global Work Area

The Global Work Area is described in member BUDWORK in the Installation Control library *?prefix.BBSAMP*.

Special Considerations for Exits

The exits must be linked into individual load module. FLST and RLST parameters are loaded into CSA or ECSA when SVOS is started or the global parameters are refreshed. If you have MVS/XA or MVS/ESA and have written the exits in 31-bit mode, the module is loaded into ECSA. The application collector does *not* have to be stopped and restarted to change any of the exits.

Since the exits may be called from application control code executing in the DADSM pre- and postprocessing exits, the following restrictions must be observed:

- runs in supervisor state in key 10
- must be coded and marked as reentrant by the linkage editor
- must be entered in 31-bit mode (MVS/XA systems only)
- cannot allocate a file dynamically
- cannot issue SVCs that ENQ on the VTOC, like LSPACE
- cannot issue SVCs that do catalog lookups that might cause a volume to be allocated

Note: If the Security Exit is used to control access to the database from the views, the security exit also must be coded to run in problem state, key 8.

In an MVS/XA 2.2 or higher environment, certain Scheduler control blocks (JCT, ACT, SCT, SIOT, JFCB) must be accessed through the use of the SWA Manager. Pointers to these control blocks may contain a SWA Virtual Address (SVA) or the TTR pointer on disk. The comments in the global work area contain the characters SVA for those fields that should use the SWA Manager. The MAINVIEW SRM Reporting component has routines to read and write SVA fields.

To use the SWA read routine, load R15 with GWASWAR and R1 with the SVA of the control block. R1 is returned with the address of a copy of the control block. The SWA write routine can be used to rewrite a modified Scheduler control block. To use this routine load the following register contents:

- R15 with GWASWAW
- R0 with the virtual address of the control block
- R1 with the SVA of the control block to be rewritten

If you are not running in an MVS/XA 2.2 environment, you may still consider using the SWA manager to provide upward compatibility.

DFHSM Exit Routines

MAINVIEW SRM DFHSM exit routines are *not* required for the reporting facility. However, if you are using DFHSM (2.4 or higher) and you want to use the MAINVIEW SRM HSMBACKP, HSMCCNV, HSMIGRT, and HSMRECAL functions, you must perform one of the following actions:

- copy the MAINVIEW SRM DFHSM exit routines (ARCMDEXT, ARCRDEXT, and ARCSAEXT) to a library where they can be accessed by DFHSM (such as a LNKST library)
- concatenate the SVOS library must into the STEPLIB for the DFHSM job(s)

After you make the MAINVIEW SRM exit routines available to DFHSM, you must issue the following command to activate them. You can issue this command from OS/390 or TSO, or you can include it in the DFHSM ARCCMDxx member in SYS1.PARMLIB.

```
SETSYS EXITON (ARCMDEXT, ARCRDEXT, ARCSAEXT)
```

To use MAINVIEW SRM DFHSM exit routines, follow these guidelines:

- If you already have DFHSM exit routines implemented at your installation, you must change the CSECT names in the MAINVIEW SRM exit routines and modify your exit routine to call the MAINVIEW SRM exit routines.

- When calling the MAINVIEW SRM DFHSM exit routines, all of the registers except register 15 should be *exactly* as they were when DFHSM called your exit routine. Register 15 should contain the entry point of the MAINVIEW SRM exit routine. This means that registers 13 and 14 return to DFHSM from the MAINVIEW SRM exit routines, not to your exit routines. The MAINVIEW SRM exit routines must receive control in 31-bit addressing mode. Any 24-bit addressing-mode user-exit routine must switch to 31-bit mode before calling the MAINVIEW SRM exit routine.
- Your exit routine must call the MAINVIEW SRM exit routines by the CSECT name that you assign instead of the standard ARCxxEXT names.

Appendix B Application Database Utilities

This appendix presents the following topics:

Overview	B-2
Initializing and Updating the Database	B-2
Control Statements	B-2
Comments	B-3
Resynchronizing the Database	B-12
SGCMAINT RESYNC Statement	B-13
SGCRSYNC Multitasking RESYNC	B-16
Diagnostic Reporting	B-17
Application Database Reports	B-17
Application Database Compare	B-20

Overview

You can use the SGCMAINT utility program to perform these tasks:

- initialize and update the database
- resynchronize the database by individual volume
- generate diagnostic reports

You can use the SGCRSYNC utility to resynchronize multiple volumes simultaneously.

Initializing and Updating the Database

The SGCMAINT utility program initializes and updates the application collector database. To ensure integrity, SGCMAINT never changes the existing database; it always creates a new database. After you verify that the new database is correct, change the SVOS started task to point to the new database and restart the application collector. This procedure provides you with the old database as a backup, if needed. Figure 2-1 contains the JCL required to run SGCMAINT. See the *MAINVIEW SRM Implementation Guide* for more information about this procedure.

Figure 2-1 SGCMAINT JCL

//INITDB	EXEC	PGM=SGCMAINT
//STEPLIB	DD	DISP=SHR,DSN=? <i>prefix</i> .BBLINK
//SYSPRINT	DD	SYSOUT=*
//SYSUDUMP	DD	SYSOUT=*
//SYSUT1	DD	UNIT=SYSALLDA,SPACE=(CYL,(1,1))
//SYSIN	DD	*

Control Statements

The following three control statements initialize the application collector database:

- File Control is required and is always the first statement. See “File Control Statement” on page B-4 for more information.

- Option Control is only required when you create a new application collector database. You can include an option on an UPDATE if you want to change any of the options, such as the size of the database. See “Option Control Statement” on page B-5 for more information.
- Field Control is the last initialization statement. One field control statement is required for each application level that is associated with a data set. See “Field Control Statement” on page B-8 for more information.

Note: The order of these statements is critical. If you cannot fit all of the parameters on a control statement, stop at the end of a keyword and repeat the INIT keyword on the next statement.

Example

This control statement
 INIT FILE(NEW) DBNAME(EMP.SGC.DB) VOLUME(MVSRES)

is the same as this statement
 INIT FILE(NEW) DBNAME(EMP.SGC.DB)
 INIT VOLUME(MVSRES)

Comments

You can use comments and blank lines to document and improve the readability of the SGCMAINT control statements. Comments should conform to the following conventions:

- Any statement with an asterisk in column 1 is considered a comment.
- Comments must be separated from the last operand on a line by at least one space followed by an asterisk.

```
*
* THIS IS A COMMENT
*
INIT FILE(NEW) DSNAME(EMP.SGC.DB)
* ANOTHER COMMENT
```

File Control Statement

The first control statement indicates whether a new database is being created or an updated copy is being created from an existing database. The format of the File Control statement follows and each keyword is explained below.

```
INIT  FILE(NEW | UPDATE)
      FROMDB(old-database)
      DBNAME(new-database)
      VOLUME(volser)
      PURGE(days-inactive)
```

FILE(NEW | UPDATE)

Required parameter. Specify **NEW** if the database is to be created. If the database options and application information records are to be copied from an existing file, specify **UPDATE**. All application information records are initialized to zero for a **NEW** file.

FROMDB(old-database)

Required when **FILE(UPDATE)** is specified. It contains the name of the application collector database to be used as input. The database application records are copied after the new application collector database control record is created. Options can be overridden in the new database control record by including option control statements.

DBNAME(new-database)

Required parameter that contains the data set name of the new application collector database. The capacity and physical size of the database is determined by the **RECORDS** parameter on the option control statement.

VOLUME(volser)

The device track capacity of the **volser** specified here or by default must be at least as great as that of a 3380. Application space utilization is tracked at the volume detail when **RESYNC** by volume is enabled. New implementations of **RESYNC** by volume would require **RESYNC** over all old volumes to establish accurate volume totals. Any partial **RESYNC** by a new implementation would invalidate the application space totals until all volumes are resynchronized.

PURGE(days-inactive)

Optional with the UPDATE option. Application information records that have not been used during the time period specified on this parameter for days-inactive are dropped from the output file. Only applications with zero space utilization are considered.

Option Control Statement

The second control statement specifies system options that are used to control resources used by an application collector, as well as the actions that can be taken when a budget is exceeded. The format of the option control statement follows and each keyword is explained below.

```
INIT  RECORDS(#-of-records)
      MODE(MONITOR | WARNING | REJECT)
      REJECT(COND | YES)
      MAXKEY(50 | max-key-length)
      HSMTRACK(YES | NO)
      HBACKUPPREFIX(DFHSM backup prefix)
      HMIGRATEPREFIX(DFHSM migrate prefix)
```

RECORDS(# of records)

Required parameter that specifies the number of database records to create. One record is required for each application. Additional records are required when applications have data sets on more volumes than is specified for RECVOLS or 902 volumes. Each additional record will accommodate the value specified for RECVOLS or at most another 902 volumes for an application. If the total number of database records is exhausted, volume totals can be lost and new applications cannot be added. The SGCMAINT program must be rerun to change the number of database records, so calculate this number as accurately as possible. The volume totals enable a RESYNC by volume. Volume totals can be recaptured by creating a copy of the database with more records and then running RESYNC against volumes with lost totals. The default value is RECORDS(1000).

RECVOLS(# volume totals/record)

Optional parameter used to implement RESYNC by volume. This keyword also specifies the number of volume space totals allowed per record for an application. A value greater than zero will collect application space totals at the volume level and enable RESYNC by volume. A zero value will disable the RESYNC by volume capability. This value is used to calculate the database record size. The maximum number of the volume totals in a database record is 902. If necessary, multiple records are used to collect all volume totals that are associated with the application.

MODE(MONITOR | WARNING | REJECT)

Optional parameter that sets the system mode option. If the system is running in MONITOR mode, the application collector only records space allocation totals. If the system is running in WARNING mode, a warning message is generated if the budget amount is exceeded. In REJECT mode, the budget amount is used to place a limit on the amount of space that can be allocated. Any allocation request that would cause an application to exceed its budget is denied. The system mode option can be overridden by using the application mode option. The application mode option can be set using the TSO BUDGET command.

REJECT(COND | YES)

Optional parameter used if MODE(REJECT) is specified. The application collector can conditionally reject an allocation request on a specific volume when you specify REJECT(COND). A code is returned to DADSM, which requests a different volume for consideration. Conditionally rejecting a DADSM request makes sense only if the volume serial number is used to determine the budget application. Otherwise, conditional rejection makes DADSM call the application collector with a new volume only to be rejected by the same application that rejected the first request. The default is YES, which causes a fail of the requested allocation.

MAXKEY(50 | max-key-len)

Optional parameter that specifies the length of the largest budget application code. While this can be calculated from the field control statements, doing so does not take into account local installation standards or application codes that are built into the selection module, Select Exit, or Key Exit.

For example, the high-level qualifier of the data set name can be up to eight characters, yet your installation standard might restrict it to five. If this was your only or longest application code, you would specify a value of 5 for MAXKEY. The default is 50 characters, which is also the maximum. The application collector builds a table in Extended CSA (for XA systems) or CSA (for 370 systems) that contains each application code and its associated record number. The application collector uses the value specified by MAXKEY to determine the amount of CSA needed for this table. If you are an MVS/370 customer, the size of this table might affect your Virtual Storage Constraints.

HSMTRACK(YES | NO)

Optional parameter that tells the application collector to track DFHSM migrate, backup, recall, and recovery functions (when YES is specified). Selection of this option causes the application collector to replace QUAL1, QUAL2, and the first two qualifiers of the DSNAME field with the values extracted from the associated migration level 0 data set name. When NO is specified, the application collector processes DFHSM requests as normal data set requests.

HBACKUPPREFIX(DFHSM backup prefix)

Required parameter if HSMTRACK (YES) is specified. HBACKUPPREFIX is used to identify DFHSM backup data sets and must specify either the UID parameter specification from the DFHSM startup procedure or the override BACKUPPREFIX specification from the DFHSM control parameter.

HMIGRATEPREFIX(DFHSM migrate prefix)

Required parameter if HSMTRACK (YES) is specified. HMIGRATE is used to identify DFHSM migrated data sets and must specify either the UID parameter specification from the DFHSM startup procedure or the override MIGRATEPREFIX specification from the DFHSM control parameters.

Unsupported Option Control Statements

The following option control statements are no longer supported. System functionality has been improved, which eliminated the need for these parameters. Informational messages will tell you that these parameters are ignored. You can remove them from your control card to eliminate the messages.

- MAXFREE—The optimum value for MAXFREE is calculated automatically each time the application collector is started.
- MAXVOLS—The maximum volume limitation has been removed.

- **MAXVOLSW**— Since there is no volume limitation, the maximum volume warning is no longer needed.
- **NOMAXVOLS**—Volume resync is always available.

Field Control Statement

The last of the three control statements (the statements that define the default application code construction rules in the system options) is the Field Control statement. A separate field control statement must be specified to activate accounting for each of the levels. During space allocation request processing, the selection module is called for each level. Use the select-language-level statements to test the level in the selection module.

Warning! *You must specify at least one level. If an INIT LEVEL parameter is not specified during database initialization, the application collector will not create or update any applications.*

The format of the field control statement follows and each keyword is explained below.

```
INIT  RECALC
      LEVEL(1 | 2 | 3 | 4)
      FIELD1(fld, pos, len)
      FIELD2(fld, pos, len)
      FIELD3(fld, pos, len)
      FIELD4(fld, pos, len)
```

RECALC

Optional parameter that explicitly defines the new application code as recalculated. All information used in defining the application code can be obtained from information already stored in the VTOC.

LEVEL(1 | 2 | 3 | 4)

Required parameter that contains a 1, 2, 3, or 4 depending on the application that is being defined. This level number can be one of the fields that make up the application code to ensure no overlap between levels. There is no hierarchical relationship between applications.

FIELD1(fld,pos,len)

Required parameter that contains the name of the first field used in creating this application code (*fld*), the position in that field (*pos*), and the number of characters to be used (*len*). Table B-1 on page B-11 contains a list of the fields that are available for building an application code and the default lengths. If you do not specify *pos*, it is assumed to be 1. If you do not specify *len*, the default length is assumed.

FIELD2(fld,pos,len)

Optional parameter that contains the second field used in creating this application code. This field is concatenated to FIELD1 to form the application code. Blanks are neither added nor removed. The description of the subfields is the same as FIELD1.

FIELD3(fld,pos,len)

Optional parameter that contains the third field used in creating this application code. This field is concatenated to FIELD2 to form the application code. Blanks are neither added nor removed. The description of the subfields is the same as FIELD1.

FIELD4(fld,pos,len)

Optional parameter that contains the fourth field used in creating this application code. This field is concatenated to FIELD3 to form the application code. Blanks are neither added nor removed. The description of the subfields is the same as FIELD1.

Examples

The following examples illustrate different approaches to initializing and updating a application database:

- In this example, the installation wants to establish a single-level the application collector application that is based on the high-level qualifier. Only 200 records are allocated because the number of high-level qualifiers is small. Warning mode is also selected. The new database is named EMP.SGC.DB and is allocated on volume ABC001.

```
INIT FILE(NEW) DBNAME(EMP.SGC.SPACEDB) VOLUME(ABC001)
INIT RECORDS(200) MODE(WARNING) MAXKEY(8)
INIT RECALC FIELD1(QUAL1) LEVEL(1)
```

- In this example, the installation wants to set up one application level to limit the total amount of space each user group can use. A user group is identified by the first three characters of the second-level qualifier of the data set name. Another application level is used to control the amount of space an individual can use. Individuals are identified by the first-level qualifier of the data set name. The number of applications anticipated is 1375. The new database is named EMP.SGC.DB and is allocated on volume ABC001.

```
INIT FILE(NEW) DBNAME(EMP.SGC.SPACEDB) VOLUME(ABC001)
INIT RECORDS(1375)
INIT MAXKEY(8) MODE(WARNING)
INIT RECALC FIELD1(QUAL2,1,3) LEVEL(1)
INIT RECALC FIELD1(QUAL1) LEVEL(2)
```

- In this example, the installation wants to establish four levels of accounting. The actual application code will be established by either the selection module or a user exit. Four levels are defined at initialization time to ensure that the select logic and user exit are used four times for each data set. The maximum application code length is 10. The installation also wants to be able to RESYNC by volume.

```
INIT FILE(NEW) DBNAME(EMP.SGC.SPACEDB) VOLUME(SUS001)
INIT RECORDS(1000) MAXKEY(10)
INIT LEVEL(1) FIELD1(QUAL1)
INIT LEVEL(2) FIELD1(QUAL2)
INIT LEVEL(3) FIELD1(QUAL3)
INIT LEVEL(4) FIELD1(QUAL4)
```

- In this example, the installation wants to change from warning to reject mode. The old database is named EMP.SGC.DB. The new database is named EMP.SGC.NEWDB and is allocated on volume XY2001. Application information will be copied from the old to the new database.

```
INIT FILE(UPDATE) FROMDB(EMP.SGC.SPACEDB)
INIT DBNAME(EMP.SGC.NEWDB) VOLUME(XY2001)
INIT MODE(REJECT)
```

- In this example, the installation wants to implement the RESYNC by Volume feature and increase the size of the database to 1000 records.

The old database is named EMP.SGC.DB. The new database is named EMP.SGC.NEWDB and is allocated on volume XY2001. Application information will be copied from the old to the new database.

```
INIT FILE(UPDATE) FROMDB(EMP.SGC.SPACEDB)
INIT DBNAME(EMP.SGC.NEWDB) VOLUME(XY2001)
INIT RECORDS(1000)
```

- In this example, the installation wants to add HSM tracking to the current database. The prefix for HSM backup data sets is HSMBKP and the prefix for migrate data sets is HSMMIG.

The old database is named EMP.SGC.DB. The new database is named EMP.SGC.NEWDB.

```
INIT FILE(UPDATE) FROMDB(EMP.SGC.SPACEDB)
INIT DBNAME(EMP.SGC.NEWDB) VOLUME(XY2001)
INIT HSMTRACK(YES)
INIT HBACKUPPREFIX(HSMBKP) HMIGRATEPREFIX(HSMMIG)
```

Application Code Fields

Table B-1 describes the fields that are available for building an application code.

Table B-1 Application Code Fields

Field Name	Length	Description
DATACLAS	08	SMS data class
DSNAME	44	data set name
DSORG	02	data set organization
LEVEL	01	application code-level number
MGMTCLAS	08	SMS management class
QUAL1	08	first-level qualifier
QUAL2	08	second-level qualifier
QUAL3	08	third-level qualifier
QUAL4	08	fourth-level qualifier
QUAL5	08	fifth-level qualifier
QUAL6	08	sixth-level qualifier
QUAL7	08	seventh-level qualifier
QUAL8	08	eighth-level qualifier
QUALL	08	last qualifier
SMS	01	SMS flag
STORCLAS	08	SMS storage class
STORGRP	08	SMS storage group
TEMP	01	temporary flag
USRC1	08	user field from the global work area
USRC2	08	user field from the global work area
VOLSER	06	volume serial number

Resynchronizing the Database

You can use the application-database resynchronization function to populate a new application database with application information or to update application information in an existing database. A production database may not contain up-to-date information for the following reasons:

- a volume was restored from a full volume backup
- a volume was removed from the system
- the system was stopped while DASD files were created, extended, renamed, released, or deleted

The RESYNC program processes each DASD volume individually. A volume is selected and all allocations to the volume are delayed while the VTOC is read and the appropriate applications are created or updated in the database.

The majority of the RESYNC elapsed time is spent performing I/O operations. RESYNC elapsed time for a volume can be roughly estimated by counting one I/O per data set and two I/Os each time a budget application is updated for a data set. Multiply the number of I/Os by the average response time for the DASD devices being used. Initially, a computation adjustment of 50 percent should be added to application for RESYNC CPU utilization and operating-system overhead. With time, you can develop a more-accurate range of computation adjustment for your site.

Individual volumes can be resynchronized with the active database. Before resynchronizing, evaluate the impact of delaying allocations on each of the volumes. Volumes with little activity probably can be resynchronized at any time with little, if any, impact. Volumes with high activity ideally should be resynchronized during a period of minimal activity to limit the impact.

The multivolume RESYNC feature of the application collector allows volume totals to be kept for each application. These totals make it possible to RESYNC individual volumes and maintain accurate application totals. If multivolume support is not selected and an attempt is made to RESYNC individual volumes, it is impossible to know what space was being tracked on these volumes prior to the RESYNC and inaccurate application totals may result.

There are two types of RESYNC: SGCMaint RESYNC and SGCRsync RESYNC.

SGCMAINT RESYNC Statement

The SGCMAINT RESYNC statement requests the resynchronization operation and specifies the database name and listing options. The format of the SGCMAINT RESYNC statement follows and each keyword is explained below.

```
RESYNC  DBNAME(database) | ACTIVEDB
        AUDIT(DSNAME, VOLSER)
        CLEAR | NOCLR
```

DBNAME(database)

Specifies the name of the application collector database to be updated.

ACTIVEDB

Updates the active application collector database (default).

Note: If you use the ACTIVEDB option, the application collector must be running. If neither DBNAME nor ACTIVEDB is specified, ACTIVEDB is assumed.

AUDIT(DSNAME, VOLSER)

Optional parameter that specifies the level of audit reporting desired. The VOLSER value generates a message for each volume. The DSNAME value generates a message for each data set. The VOLSER value is recommended, especially if your installation wants to exclude certain volumes from application collector control. Either one or both values may be specified.

CLEAR | NOCLR

Optional parameter that controls whether the current allocation totals are to be cleared before starting the RESYNC operation. If RESYNC by volume is in effect, the volume totals are automatically cleared for each volume to be resynchronized, and current allocation totals are recalculated. If RESYNC by volume is not in effect, current allocation totals are cleared by default.

The CLEAR | NOCLR parameter allows the RESYNC operation to be divided into several steps. If RESYNC by volume is in effect and a volume is being RESYNCEd for the first time, new volume-total entries are built and current space allocation totals are recalculated. If RESYNC by volume is not in effect, a RESYNC of all of the DASD can be divided into multiple steps with CLEAR specified in the first step and NOCLR specified in subsequent steps. This action prevents the current allocation totals from being cleared unnecessarily in each step.

VOLUME Selection Statement

VOLUME Selection statements are optional and restrict the resynchronization operation to specific volume serials. Use the INCLUDE and EXCLUDE statements to specify the volumes to be included or excluded. The INCLUDE and EXCLUDE statements cannot be mixed in a RESYNC operation. You must use either all INCLUDE or all EXCLUDE statements. Use the CLEAR statement to remove totals for specific volumes from the application collector database.

The volume serial numbers specified on these statements can use masking. Code only one volume serial number or a mask. The total number of selection statements cannot exceed 40. The format of this control statement is as follows:

```
VOLUME INCLUDE(volser)
VOLUME EXCLUDE(volser)
VOLUME CLEAR(volser)
```

INCLUDE(volser)

Specifies a volume serial number or a mask that the application collector considers during the RESYNC operation. Volume serial numbers that are not specified or that do not match a specified mask are not used to rebuild the application collector database. For example, if you use VOLUME INCLUDE(PROD/,WORK/), the system finds only those volumes that meet the PROD/ mask and ignores the WORK/ mask. To code both, use separate INCLUDE statements.

EXCLUDE(volser)

Specifies a volume serial number or a mask that the application collector does not consider during the RESYNC operation. The volumes that are not specified or that do not match the specified mask are used to rebuild the application collector database.

CLEAR(volser)

Specifies a volume serial or mask that the application collector uses in the RESYNC operation to remove totals for specific volumes from the application collector database. Use this statement only if you have selected multivolume support. The CLEAR statement can be combined with INCLUDE or EXCLUDE statements.

Examples

The following examples illustrate several RESYNC scenarios:

- In this example, the installation wants to create an application database and then populate it with current information by running RESYNC. RESYNC by volume is in effect and the application collector is not active. Activity on the system is minimal. This is the fastest way to build a database with the least impact on end users.

```
RESYNC DBNAME(EMP.SGC.TEST)
```

- In this example, the installation wants to RESYNC the active database. The application collector is active and RESYNC by volume is in effect.

```
RESYNC ACTIVEDB
```

- In this example, the installation wants to RESYNC a group of volumes whose volume serial number begins with PROD with the active database. The application collector is active and RESYNC by volume is in effect.

```
RESYNC ACTIVEDB  
VOLUME INCLUDE(PROD/)
```

- In this example, the installation wants to clear the totals for those volumes with a volume serial starting with TESTA from the application collector database. The application collector is active and RESYNC by volume is in effect.

```
RESYNC ACTIVEDB  
VOLUME CLEAR(TESTA/)
```

- In this example, the installation wants to completely RESYNC the active database and list each volume.

```
RSYNC ACTIVEDB AUDIT(VOLSER)
```

- In this example, the installation wants to completely RESYNC the active database, list each volume, and list data set names on volume EMP001.

```
RSYNC ACTIVEDB AUDIT(DSNAME,VOLSER)  
VOLUME INCLUDE (EMP001)
```

SGCRSYNC Multitasking RESYNC

Use SGCRSYNC to resynchronize multiple volumes simultaneously, which is much faster than SGCMAINT RESYNC. Member SGCMTJR1 in *?prefix*.BBSAMP contains sample JCL for SGCRSYNC. SGCRSYNC keywords are as follows:

RESYNC

Select volumes to be resynchronized. Multiple RESYNC statements can be specified.

INCLUDE()/EXCLUDE()/ALL VOLS

Used to select volumes; can include or exclude volumes using either a *single* volume serial or a *single* mask. For example, if you code RESYNC INCLUDE(BSD/,EMP/), the system finds only those volumes that start with the BSD/ mask and ignores the EMP/ mask. To code both, use separate INCLUDE statements. EXCLUDE control statements must appear after all INCLUDE control statements.

CLEAR

Used to clear volume totals in the database. Use this parameter to clear volume specific totals when the volume no longer exist on the system. The default is ALLVOLS.

MAXTASKS

Controls the number of RESYNC tasks.

TASKS()

Specify the number of RESYNC tasks. The RESYNC tasks run simultaneously. Each task will RESYNC one volume at a time until all the selected volumes have been resynchronized. The default is 1 and the maximum is 16.

RESERVE

Optional, controls use of volume reserves. The default value is ALLVOLS. Multiple RESERVE statements may be specified.

INCLUDE()/EXCLUDE()/ALL VOLS

Used to select volumes. Can include or exclude volumes using either a single volume serial or mask.

AUDITDSN

Displays application information for each data set.

AUDITVOL

Displays status for each volume.

Examples of Control Statements

The following examples illustrate several RESYNC control statements:

- RESYNC all volumes using three RESYNC tasks.

```
RESYNC  
MAXTASKS TASKS(3)
```

- RESYNC all volumes with volume serials starting with EMP. Three RESYNC tasks are used.

```
RESYNC INCLUDE (EMP/)  
MAXTASKS TASKS(3)
```

- RESYNC all volumes with volume serials starting with EMP except EMP001. Ten RESYNC tasks are used.

```
RESYNC INCLUDE(EMP/)  
RESYNC EXCLUDE(EMP001)  
MAXTASKS TASKS(10)
```

Diagnostic Reporting

You can use the SGCMAINT batch utility program to print database records and to compare databases.

Application Database Reports

Use the SGCMAINT batch utility program to print database records through the use of the REPORT statement. Specify the name of the application collector database in the DBNAME field of the Report Control statement. The two formats of the REPORT statement follow and each keyword is explained below.

```
REPORT FORMAT(DBC | DBI) DBNAME(database)
REPORT DUMP(DBC | DBI) DBNAME(database)
```

FORMAT(DBC | DBI)

Optional parameter that specifies that either the database control record (DBC) or each application information record (DBI) is to be printed with each field formatted.

DUMP(DBC | DBI)

Optional parameter that specifies that either the database control record (DBC) or each application information record (DBI) is to be printed in hexadecimal characters.

DBNAME(database)

Optional parameter that specifies the name of the application collector database to be printed. The active database is assumed if DBNAME is omitted.

REPORT Examples

The following examples and sample output illustrate several REPORT functions:

- In this example, system options contained in the database control record from the active database are formatted and printed.

```
REPORT FORMAT(DBC)
```

Figure B-2 Sample Output - Example 1

```

SGC5400 TOTAL # OF SLOTS: 1000, USED SLOTS: 196
SGC5401 ACCOUNT RULE INFORMATION FOLLOWS FOR LEVEL 1
SGC540A ACCOUNT CODE INFORMATION WILL BE DYNAMICALLY RECALCULATED
SGC5403 THE ACCOUNT IS CONSTRUCTED USING THE FOLLOWING INFORMATION
SGC5404 FIELDNAME: QUAL1, POSITION: 1, LENGTH: -COMPLETE-
SGC5401 ACCOUNT RULE INFORMATION FOLLOWS FOR LEVEL 2
SGC540A ACCOUNT CODE INFORMATION WILL BE DYNAMICALLY RECALCULATED
SGC5403 THE ACCOUNT IS CONSTRUCTED USING THE FOLLOWING INFORMATION
SGC5404 FIELDNAME: QUAL1, POSITION: 1, LENGTH: -COMPLETE-
SGC5404 FIELDNAME: QUAL2, POSITION: 1, LENGTH: -COMPLETE-
SGC5401 ACCOUNT RULE INFORMATION FOLLOWS FOR LEVEL 3
SGC5409 NO ACCOUNT CODE CONSTRUCTION INFORMATION FOUND AT THIS LEVEL
SGC5401 ACCOUNT RULE INFORMATION FOLLOWS FOR LEVEL 4
SGC5409 NO ACCOUNT CODE CONSTRUCTION INFORMATION FOUND AT THIS LEVEL
SGC5405 SPACE INFORMATION IS KEPT AS KBYTES
SGC5406 REQUESTS WILL BE UNCONDITIONALLY REJECTED
SGC5407 SYSTEM IS IN WARNING MODE
SGC5408 MAXIMUM ACCOUNT CODE LENGTH: 16
SGC540J MAXIMUM NUMBER OF VOLUMES SUPPORTED: 0
SGC540K WARNING MESSAGE AFTER 80 % OF VOLUME ENTRIES HAVE BEEN USED
SGC540L TRACK DFHSM REQUESTS: YES
SGC540M USE PART OR ALL OF THE ML0 DSN: PART
SGC540N MIGRATE DATASET PREFIX: HSM26, BACKUP DATASET PREFIX: HSM26

```

- In this example, application records from a named database are formatted and printed.

```
REPORT FORMAT(DBI) DBNAME(<DBHLQ>).SGC.SPACEDB)
```

Figure B-3 Sample Output - Example 2

```

SGC540B ACCOUNT: CONTROL - RECORD ID: 1
SGC540C INFO FOLLOWS FOR SPACE TYPE PERMANENT
SGC540G MAX: 999999 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540C INFO FOLLOWS FOR SPACE TYPE TEMPORARY
SGC540G MAX: 999999 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540C INFO FOLLOWS FOR SPACE TYPE VSAM
SGC540G MAX: 999999 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540C INFO FOLLOWS FOR SPACE TYPE HSM
SGC540G MAX: 0 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540D WARNING PERCENTAGE: 80% - ACCOUNT: 'SG-Control DEFAULT' CREATED:
12/12/91 - LAST USED:

SGC540B ACCOUNT: TEST - RECORD ID: 2
SGC540C INFO FOLLOWS FOR SPACE TYPE PERMANENT
SGC540G MAX: 999999 - CURRENT: 19186 - HIGH WATER MARK: 19186
SGC540C INFO FOLLOWS FOR SPACE TYPE TEMPORARY
SGC540G MAX: 999999 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540C INFO FOLLOWS FOR SPACE TYPE VSAM
SGC540G MAX: 999999 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540C INFO FOLLOWS FOR SPACE TYPE HSM
SGC540G MAX: 0 - CURRENT: 0 - HIGH WATER MARK: 0
SGC540D WARNING PERCENTAGE: 80% - ACCOUNT: 'SG-Control DEFAULT' CREATED:...

```

Application Database Compare

You can use the SGCMAINT batch utility program to compare two databases. This may be useful after a RESYNC to populate a new version of the database. Each record is read sequentially from the TODB and a random read is made to the FROMDB. While this is generally sufficient to detect any differences between the two files, it does not detect a record in the FROMDB that is not in the TODB. You can run the compare both ways to be absolutely certain all differences are discovered.

All application information record (DBI) fields are compared except *date of last use* and *creation date*. These fields are updated on a RESYNC if the budget application is updated. The format of the COMPARE statement follows and each keyword is explained below.

```
COMPARE FROMDB(random-read-db)
COMPARE TODB(sequential-read-db)
```

FROMDB(random-read-db)

Required parameter that specifies the input database to be read randomly.

TODB(sequential-read-db)

Required parameter that specifies the input database to be read sequentially.

Example

The following example illustrates a COMPARE function in which two databases are compared.

```
COMPARE FROMDB(EMP.SGC.SPACEDB)
COMPARE TODB(EMP.SGC.TEST)
```

Index

Symbols

* parameter
BUDDSN command 5-20

A

access control
 security exit A-5
Access Method Services 8-3
acct-code parameter
 BUDGET command 5-22, 5-26, 5-32
action parameters
 graphic overview 6-23
 output management 6-21
ACTIVEDB parameter
 RESYNC statement B-13
ACTLOGTYPE(DASD) parameter 6-16
ACTLOGTYPE(SYSOUT) 6-16
add exit A-2, A-9
 register conventions A-9
add processing 5-4
ADD/CHANGE parameter
 BUDGET command 5-26
allocate from pool during DFHSM recall 2-31
AMS 8-3
application
 reviewing 5-8
 updating 5-8
application code
 build exit A-2, A-7
 BUDKEY01 A-15

 register conventions A-8
 data sets 5-8
 definition 5-6
 determining 5-8
 lengths B-11
 types B-11
application collector
 add processing 5-4
 check processing 5-2
 functionality 5-2
 use with DADSM exits 5-2
application database
 initializing B-2
 reports B-17
 updating B-2
applications
 creating 5-5
 TSO BUDGET COMMAND 5-6
apply calendar conversion to DFHSM migration
 2-23
asterisk
 comment line 2-42, 6-25
 name mask 2-42
ATL 8-3
AUDIT parameter
 RESYNC statement B-13
auditing
 COMPARE function B-20
Automatic Tape Libraries 8-3

B

BACKCMD 2-18

BACKUP 2-18, 2-27, 6-26
 backup
 controlling for data sets under DFHSM 2-17
 report 5-11
 backup log 6-18, 6-26
 backups, control DFHSM 2-17
 BAKDAYS 2-36, 2-37
 batch processing 6-5
 batch reporting utilities
 BUDGET TSO command 5-10
 COBOL programs 5-10
 DFHSM migration and backup 5-10
 SAS procedures 5-10, 5-12
 batch utilities
 initializing a database B-2
 SGCMAINT B-2
 updating a database B-2
 batch utility, SMHSMUTL - DFHSM control
 data set 2-36
 BBPARM members
 SMFLSTPxx 4-6
 SMFUNCxx 4-6
 SMMSYSxx 3-6
 BBSAMP
 sample exits A-15
 sample programs 5-10
 BBSAMP members
 BUDWORK A-18
 DBI41 A-16
 SGC1CMLK 5-13
 SGC2CMLK 5-13
 SGCCOB02 5-13
 SGCCOB03 5-17
 SGCCOBJ2 5-17
 SGCCOBJ3 5-17
 SGCHSMJ1 5-11
 SGCMTRJ1 B-16
 SGCSAS01 5-12
 SGCSAS02 5-12
 SGDCOLLS 3-8
 SGDCOPYJ 3-3
 BCDS 2-42
 BCDSn 6-9, 6-10
 BMC
 CONTROL-T 8-3
 BUDCHK01
 check exit A-15
 BUDDDEF01
 default exit A-15
 BUDDSN command 5-18
 * parameter 5-20
 application codes, single data set 5-18
 data set parameter 5-18
 listing application codes for data sets on
 volume 5-20
 listing application codes, group of data sets
 5-19
 listing budget applications 5-18
 masking 5-19
 pattern parameter 5-19
 single data set 5-18
 VOLSER parameter 5-20
 BUDGET command 5-21
 acct-code parameter 5-22, 5-26, 5-32
 add/change multiple application codes 5-30
 ADD/CHANGE parameter 5-26
 add/change single budget application code
 5-26
 CHANGE/ADD parameter 5-26
 CURPERM parameter 5-32
 CURTEMP parameter 5-32
 CURVSAM parameter 5-32
 DEFAULT parameter 5-28
 Field Table 5-25
 field1 subparameter 5-23
 field2 subparameter 5-24
 FROM parameter 5-26
 HWMPERM parameter 5-33
 HWMTEMP parameter 5-33
 HWMVSAM parameter 5-33
 LIST parameter 5-22
 list/change budget applications 5-21
 listing information, multiple application
 codes 5-23
 listing information, single budget
 application code 5-22
 MAXPERM parameter 5-27
 MAXTEMP parameter 5-27
 MAXVSAM parameter 5-27
 MONITOR parameter 5-28
 NAME parameter 5-26
 NOHSM parameter 5-28
 NOTEMP parameter 5-27
 NOVSAM parameter 5-27
 operator subparameter 5-23
 PHSM parameter 5-28

PTEMP parameter 5-27
 PVSAM parameter 5-27
 REJECT parameter 5-28
 RESET parameter 5-28
 THSM parameter 5-28
 TOTAL parameter 5-23
 TTEMP parameter 5-27
 TVSAM parameter 5-28
 USERF1 parameter 5-26
 USERF2 parameter 5-26
 USERF3 parameter 5-27
 value subparameter 5-23
 WARN parameter 5-28
 WARNING parameter 5-28
 budget models
 default exit A-12
 BUDGET TSO command
 JCL required 5-11
 BUDKEY01
 application code build exit A-15
 BUDSAS 5-12
 BUDSAS01
 SAS reporting procedure 5-12
 BUDSAS02
 SAS reporting procedure 5-12
 BUDSEC01
 security exit A-15
 BUDSEC02
 security exit A-15
 BUDSEC03
 security exit A-15
 BUDSEC04
 security exit A-15
 BUDSEL01
 select exit A-15
 BUDSEL02
 select exit A-15

C
 CA
 CA-1 8-3
 CA-1 8-3, 8-4
 CAL 2-23, 2-27, 2-28, 2-36, 2-37
 CALAGE 2-21, 2-29
 calendar
 conversion 2-23
 calendar conversion to DFHSM migration, apply
 2-23
 cards, control 6-14
 CAT 2-21, 2-29, 2-33
 CDS 2-36, 2-38
 CHANGE/ADD parameter
 BUDGET command 5-26
 check exit A-2, A-10
 BUDCHK01 A-15
 check processing 5-2
 CLEAR B-14
 CLEAR/NOCLR parameter
 RESYNC statement B-13
 CMF parameters 4-2
 COBOL reporting program
 SGCCOB02 5-17
 SGCCOB03 5-17
 coding rules
 SGCMAINT control statements B-3
 system exits A-18
 TSO commands 5-22
 COMMAND 6-26
 command log 6-18, 6-26
 commands
 ENDTSCAN 8-5
 TSCAN 8-5
 COMPARE statement B-20
 after change B-20
 after RESYNC B-20
 for auditing B-20
 TODB parameter B-20
 comparing
 database records B-20
 Computer Associates
 Tape Management System 8-3
 control
 migration 2-21, 2-27
 control cards 6-14
 control data set
 batch utility
 SMHSMUTL 2-36
 control data sets
 DFHSM 6-2
 control DFHSM backups 2-17
 control statements
 job 6-6
 output management 6-21
 CONTROL-T 8-4

- Tape Management System 8-3
- conversion to DFHSM migration, apply calendar 2-23
- copy/merge utility
 - keywords 3-4
 - processing options 3-3
- creating applications 5-5
- CREDAYS 2-36, 2-38
- CURPERM parameter
 - BUDGET command 5-32
- CURTEMP parameter
 - BUDGET command 5-32
- CURVSAM parameter
 - BUDGET command 5-32

D

- DADSM
 - post-processing exit 5-2
 - pre-processing exit 5-2
 - use with application collector 5-2
- DADSM exits A-18
- DASD
 - reporting space availability 5-9
 - space usage 5-2
 - storage management 5-2
- data collection
 - multiple sources 8-3
- data control
 - security exit A-5
- data definitions
 - SGCDB 3-9
 - SGRDACNT 3-9
 - SGRDDSN 3-9
 - SGRDPOOL 3-8
 - SGRDVOL 3-8
 - SIBLMSG 3-9
 - SIBRMSG 3-9
 - STEPLIB 3-9
 - SVWEXEC 3-9
 - SYSIN 3-9
 - SYSPRINT 3-9
 - SYSTEMM 3-9
 - SYSTSPRT 3-9
- data set parameter
 - BUDDSN command 5-18
- data sets

- batch utility, SMHSMUTL - DFHSM 2-36
- cleanup utility 2-36
- created by output management 6-17
- DFHSM backup 2-17
- DFHSM control 6-2
- during DFHSM recall 2-31
- list 6-34
- migration control 2-21, 2-27
- output 6-17
- preventing simultaneous updates 3-11
- recall control 2-31
- security 3-11
- size 2-31
- DBNAME parameter
 - file control statement B-4
 - REPORT statement B-18
- ddname descriptions 6-21
- default application code construction rules
 - definition 5-8
- default exit A-2
 - BUDDEF01 A-15
 - budget models A-12
 - uses A-12
- DEFAULT parameter
 - BUDGET command 5-28
- defining applications 5-18
- definition members
 - SMFUNCxx 2-2
- DELETE 2-36, 2-37
 - parm on TSO BUDGET Command 5-29
- descriptions, ddname 6-21
- DEVTYPE 2-33
- DFDSS 6-26
- DFHSM
 - backups, control 2-17
 - control data set 6-2
 - batch utility
 - SMHSMUTL 2-36
 - exits A-19
 - migration and backup report 5-11
- DFHSM enhancements
 - HSMBACKUP, controls DFHSM data set
 - backup 2-17
 - HSMDELETE, improves data set deletion 2-21
 - HSM MIGRT, improves data set migration 2-27

HSMRECAL, controls pooling during
DFHSM recall process 2-31
SMHSMUTL, controls data set cleanup
utility 2-36
DFHSM environment
defining to MAINVIEW SRM 6-2
DFHSM log files 6-3
extracting records 6-3
DFHSM migration, apply calendar conversion to
2-23
DFHSM recall, allocate from pool during 2-31
DFHSMShsm 8-4
DFSMSdfp 8-3, 8-4
DFSMSShsm 8-3
DFSMSrmm 8-3, 8-4
DSN 2-11, 2-14, 2-21, 2-29, 2-33, 2-36, 2-38
DSNn 2-21, 2-29, 2-33
DSORG 2-21, 2-29, 2-33
DSTYPE 2-11, 2-14, 2-21, 2-29
DUMP 6-26
log 6-18, 6-26
DUMP parameter
REPORT statement B-18
during DFHSM recall, allocate from pool 2-31

E

EasyHSM functions
HSMCCNV 2-23
HSMRECAL 2-31
SMHSMUTL 2-36
ENDTSCAN command 8-5
EVENTID 2-4, 2-5, 2-9, 2-10, 2-11, 2-12, 2-13,
2-14, 2-18, 2-19, 2-21, 2-23, 2-24, 2-27, 2-28,
2-32, 2-36, 2-39
examples
BUDGET command 5-24, 5-31, 5-33
COMPARE operation B-20
REPORT operation B-18
RESYNC function B-15
symbolic substitution 2-7
EXCLUDE B-14
VOLUME Selection statement B-14
exits
coding rules A-18
flow A-3
restrictions A-18

system exits A-3

F

field control statement B-8
FIELDn parameter B-9
LEVEL parameter B-8
RECALC parameter B-8
field layout
global work area A-18
field table
BUDGET command 5-25
field1 subparameter
BUDGET command 5-23
field2 subparameter
BUDGET command 5-24
FIELDn parameter
field control statement B-9
file control statement B-4
DBNAME parameter B-4
FILE parameter B-4
FROMDB parameter B-4
PURGE parameter B-5
VOLUME parameter B-4
FILE parameter
file control statement B-4
files, DFHSM log 6-3
filtering 6-14
filtering, message 6-15
filters
messages 6-15
FORMAT parameter
REPORT statement B-18
FROM parameter
BUDGET command 5-26
FROMDB parameter
COMPARE statement B-20
file control statement B-4
functions
HSMBACKP 2-17
HSMDELET 2-21
HSMCCNV 2-23, A-19
HSMMIGRT 2-27, A-19
HSMRECAL 2-31, A-19
SGPCCURC
cache controller records 2-13
SMHSMUTL 2-36

G

GDGVER 2-21, 2-29
global parameters 8-6
 StorageGUARD 3-6
global work area A-18
 description A-18
 record layout A-18

H

HBACKUPPREFIX parameter
 option control statement B-7
high-water marks
 definition 5-6
historical performance 4-1
 global parameters 4-5
historical performance reporting
 SMF and CMF/RMF parameters 4-2
historical space
 data 3-1
HIT operand
 use of 2-7
HLOGAUTH 6-10
HLOGAUTM 6-11
HLOGCOLL 6-11
HLOGINDX 6-8, 6-12
HLOGPRIM 6-12
HLOGTASK 6-13
HLOGUNIT 6-13
HLOGYDSN 6-14
HLQ 2-21, 2-29, 2-33
HMIGRATEPREFIX parameter
 option control statement B-7
Host Software Component 8-3
HSC 8-4
 StorageTek 8-3
HSMACTID 6-16
HSMBACKP 2-17
HSMCMD 2-42
HSMDELET 2-21
HSMGCCNV 2-23
HSMIGRT 2-27
 selection parameters 2-29
HSMRECAL 2-31
 selection parameters 2-33
HSMTRACK parameter

 option control statement B-7
HWMPERM parameter
 BUDGET command 5-33
HWMTEMP parameter
 BUDGET command 5-33
HWMVSAM parameter
 BUDGET command 5-33

I

IBM 8-3
 Access Method Services 8-3
 DFSMSdfp 8-3
 DFSMSHsm 8-3
 DFSMSRmm 8-3
 OAM 8-3
 SMF 8-3
IGGPOST00
 DADSM post-processing exit 5-2
IGGPREE00
 DADSM pre-processing exit 5-2
INCLUDE parameter B-14
 VOLUME Selection statement B-14
INIT statements
 SGCMAINT control statements B-3
initialization
 of the application database B-2
 SGCMAINT B-2
INTRDR 6-20, 6-21

J

JCL
 BUDGET TSO command 5-11
 SGCMAINT B-2
JCLOPM 6-20, 6-21
JOB 2-10, 2-11, 2-12, 2-13, 2-14, 2-33
job control statements 6-6
job submission 6-14
JOBACCTn 2-33
JOBCLASS 2-33

K

KEEPBACKV 2-36, 2-39
keyword operand

NEWAPPL statement 2-7

L

LEVEL parameter
field control statement B-8
LIST parameter
BUDGET command 5-22
list, data set 6-34
list, member 6-34
listing applications 5-18
LLQ 2-21, 2-29, 2-33
log
backup 6-18, 6-26
command 6-18, 6-26
migration 6-18, 6-26
log files
DFHSM 6-3
extracting records 6-3

M

MAINVIEW SRM
historical performance data 4-1
system parameters 6-8
masking B-14
BUDDSN command 5-19
name 2-42, 6-25
MAXKEY parameter
option control statement B-6
MAXPERM parameter
BUDGET command 5-27
MAXTEMP parameter
BUDGET command 5-27
MAXVSAM parameter
BUDGET command 5-27
MCDS 2-42
MCDSn 6-9
member list 6-34
messages
filtering 6-15
HSMBACKP 2-19
HSMDELET 2-22
HSMGCCNV 2-25
HSMIGRT 2-30
HSMRECAL 2-34

message data set processing 6-26
SMHSMUTL 2-42
MGMTCLAS 2-24
MIGCMD 2-27, 2-28
MIGDAYS 2-36, 2-39
MIGRATE 2-27, 2-28
MIGRATION 6-26
migration
apply calendar conversion to DFHSM 2-23
control 2-21, 2-27
log 6-18, 6-26
ML2 2-27, 2-29
mode
active 2-19, 2-22, 2-25, 2-30, 2-34, 2-41
MODE parameter
option control statement B-6
MONITOR parameter
BUDGET command 5-28
multiple space data collectors
simultaneous updates 3-12
multitasking RESYNC
SGCRSYNC B-16
multi-volume RESYNC B-12

N

name masking 2-42, 6-25
NAME parameter
BUDGET command 5-26
NEWACCT
see NEWAPPL 2-4
NEWAPPL 2-4
defined 2-5
keyword operand 2-7
text operand 2-6
NOCLR/CLEAR parameter
RESYNC statement B-13
NODELETE
parm on TSO BUDGET command 5-29
NOHSM parameter
BUDGET command 5-28
NOTEMP parameter
BUDGET command 5-27
NOVSAM parameter
BUDGET command 5-27

O

- OAM 8-3, 8-4
- Object Access Method 8-3
- online views 6-14
- operator subparameter
 - BUDGET command 5-23
- OPMIN 6-21
- OPMSKL 6-21
- OPMSUB 6-21
- OPRTR000 program 6-16
- option control statement B-5
 - HBACKUPPREFIX parameter B-7
 - HMIGRATEPREFIX B-7
 - HSMTRACK parameter B-7
 - MAXKEY parameter B-6
 - MODE parameter B-6
 - RECORDS parameter B-5
 - RECVOLS parameter B-6
 - REJECT parameter B-6
- output data sets 6-17
- output management 6-34
 - naming conventions for input data sets 6-16
 - output data sets 6-17
 - parameters for DFHSM and DFDSS 6-24
- output parameters 6-27
- overriding applications 5-18

P

- parameters
 - output 6-27
 - system 6-8
 - use in output management 6-24
- pattern parameter
 - BUDDSN command 5-19
- percent sign
 - skeleton processing 6-29
- performance A-5, A-18
- PGM 2-33
- PGMRNAME 2-33
- PHSM parameter
 - BUDGET command 5-28
- POOL 2-12, 2-21, 2-29, 2-32, 2-36, 2-40
- pool during DFHSM recall, allocate from 2-31
- processing
 - add 5-4

- batch 6-5
- check 5-2
- skeleton 6-28
- PROCSTEP 2-33
- PTEMP parameter
 - BUDGET command 5-27
- PURGE parameter
 - file control statement B-5
- PVSAM parameter
 - BUDGET command 5-27

R

- RACF 2-21, 2-29, 2-33
- RECALC parameter
 - field control statement B-8
- recall
 - allocate from pool during DFHSM 2-31
- recalls
 - controlling pooling during 2-31
- record layout
 - application record A-16
 - global work area A-18
- RECORDS parameter
 - option control statement B-5, B-6
- REFAGE 2-21, 2-29
- REFDAYS 2-36, 2-40
- register conventions
 - add exit A-9
 - application code build exit A-8
 - security exit A-6
- REJECT parameter
 - BUDGET command 5-28
 - option control statement B-6
- REPORT statement B-17
 - DBNAME parameter B-18
 - DUMP parameter B-18
 - FORMAT parameter B-18
- reporting
 - database records B-17
 - DFHSM migration and backup 5-11
 - diagnostic B-2
 - trend analysis 5-10
- RESET parameter
 - BUDGET command 5-28
- RESYNC
 - functionality 5-6

- multivolume feature B-12
- SGCRSYNC B-12
- SGMAINT B-12
- single pack restore B-14
- statement B-13
 - ACTIVEDB parameter B-13
 - AUDIT parameter B-13
 - CLEAR/NOCLR parameter B-13
- resynchronizing
 - database B-12
 - reasons for B-12
- RLS 6-14
- RMF parameters 4-2

S

- SAS reporting procedure
 - batch 5-12
 - BUDSAS01 5-12
 - BUDSAS02 5-12
- SCAN 2-4
- security
 - data sets 3-11
 - StorageGUARD space data collector 3-11
- security considerations
 - TSO commands 5-18
- security exit A-2
 - access control A-5
 - BUDSEC01 A-15
 - BUDSEC02 A-15
 - BUDSEC03 A-15
 - data control A-5
 - performance A-5
 - register conventions A-6
 - uses A-4, A-7, A-9, A-10, A-12, A-13
- select exit A-2, A-13
 - BUDSEL01 A-15
 - BUDSEL02 A-15
- selection parameters
 - graphic overview 6-23
 - output management 6-22
- SETSRM keywords
 - SGDID 3-10
- SG_INITPOOL 3-7
- SG_INITVOL 3-7
- SG_MAXACCT 3-7
- SG_READNTVL 3-7
- SG_RETRYLIM 3-7
- SG_SPACHLDR 3-7
- SG_SUBTASKS 3-7
- SG_WRITNTVL 3-7
- SGCCOB02
 - COBOL reporting program 5-17
- SGCCOB03
 - COBOL reporting program 5-17
- SGCMAINT
 - COMPARE function B-20
 - COMPARE statement B-20
 - control statements
 - sequence of B-2
 - field control statement B-8
 - initialization B-2
 - JCL required B-2
 - option control statement B-5
 - REPORT function B-17
 - REPORT statement B-17
 - RESYNC statement B-13
 - VOLUME Selection statement B-14
- SG-Control
 - use with DADSM exits 5-2
- SGCRSYNC
 - described B-16
 - RESYNC B-12
- SGD_PROCNM 3-7
- SGD_SMFID 3-7
- SGDA 2-14
 - SGDA_ALNV 2-9, 2-14
 - SGDA_ALV 2-9, 2-14
 - SGDA_AVAIL 2-9, 2-14
 - SGDA_GRP 2-9, 2-14
 - SGDA_IDLE 2-9, 2-14
 - SGDA_NVDS 2-9, 2-14
 - SGDA_VDS 2-9
 - SGDA_VSD 2-14
- SGDCOLLECT 3-7, 3-8
- SGDCOLLECTn 3-7
- SGDP_ALNV 2-9, 2-14
- SGDP_ALV 2-9, 2-14
- SGDP_AVAIL 2-9, 2-14
- SGDP_IDLE 2-9, 2-14
- SGDP_NCLPER 2-9, 2-14
- SGDP_NNV 2-9, 2-14
- SGDP_NV 2-9, 2-14
- SGDP_NVOL 2-9, 2-14
- SGDP_PERFUL 2-9, 2-14

SGDP_POOL 2-9, 2-14	SGP_CFWHIT@ 2-13, 2-15
SGDP_RSVD 2-9, 2-14	SGP_CFWPRSC 2-13, 2-15
SGDP_RVAARC 2-9, 2-14	SGP_CHPID 2-11, 2-15
SGDP_RVAFNC 2-9, 2-14	SGP_CNTLUID 2-10, 2-16
SGDP_RVAFSC 2-9, 2-14	SGP_CONNTIM 2-11, 2-12, 2-13, 2-16
SGDP_RVAIND 2-9, 2-14	SGP_CUBSYDL 2-13, 2-16
SGDP_RVANCL 2-9, 2-14	SGP_DFWHIT@ 2-13, 2-16
SGDP_TYPE 2-9, 2-15	SGP_DFWPRSC 2-13, 2-16
SGDPROCNMn 3-7	SGP_DISCTIM 2-11, 2-12, 2-13, 2-16
SGDSMFIDn 3-7	SGP_DP@BUSY 2-11, 2-12, 2-16
SGDV_ALREXT 2-10	SGP_DPBSYDL 2-13, 2-16
SGDV_ALREXT=nnnnn 2-15	SGP_DSALLOC 2-13
SGDV_FRAGI 2-10, 2-15	SGP_DSNCNT 2-13
SGDV_FRCYL 2-10, 2-15	SGP_DVBSYDL 2-13, 2-16
SGDV_FREXT 2-10, 2-15	SGP_ECMCFBS 2-12, 2-16
SGDV_FRVIR 2-10, 2-15	SGP_ECMMSGs 2-12, 2-16
SGDV_IDTR 2-10, 2-15	SGP_ECMNSPC 2-12, 2-16
SGDV_LREXT 2-10, 2-15	SGP_ECMPGMS 2-12, 2-16
SGDV_LREXTT 2-10, 2-15	SGP_FSCBYRD 2-12, 2-16
SGDV_NDS 2-10, 2-15	SGP_FSCPERC 2-12, 2-16
SGDV_NF0DSC 2-10, 2-15	SGP_FSUPERC 2-12, 2-16
SGDV_PERFUL 2-10, 2-15	SGP_IOPRSEC 2-10, 2-11, 2-12, 2-13, 2-16
SGDV_POOL 2-10, 2-15	SGP_IOSQTIM 2-11, 2-12, 2-13, 2-16
SGDV_POOL1 2-10, 2-15	SGP_JOBcnt 2-11
SGDV_PTYP 2-10, 2-15	SGP_LCU@BSY 2-12
SGDV_RSRVDT 2-10, 2-15	SGP_LCU@BUSY 2-16
SGDV_RVAFDV 2-10, 2-15	SGP_LCUID 2-12, 2-16
SGDV_RVAIND 2-10, 2-15	SGP_MAXCCUS 4-6
SGDV_RVAPCS 2-10, 2-15	SGP_MAXDIRS 4-6
SGDV_RVAPCU 2-10, 2-15	SGP_MAXDSN 4-5
SGDV_RVASSF 2-10, 2-15	SGP_MAXJOB 4-5
SGDV_RVAVOL 2-10, 2-15	SGP_MAXLCUS 4-6
SGDV_USEXT 2-10, 2-15	SGP_MAXPOLs 4-5
SGDV_VOL 2-10, 2-15	SGP_MAXPTHs 4-5
SGINITPOOLn 3-7	SGP_MAXPVLS 4-6
SGINITVOLn 3-7	SGP_MAXSCL 4-5
SGMAINT	SGP_MAXVOLS 4-5
RESYNC B-12	SGP_NCL 2-12
SGMAXACCTn 3-7	SGP_NCLPERC 2-12, 2-16
SGMAXPOOLn 3-7	SGP_NRDHIT 2-10
SGMAXSSDSZn 3-7	SGP_NRDHIT@ 2-13, 2-16
SGP_@BUSY 2-13, 2-15	SGP_NRDpSEC 2-10, 2-13, 2-16
SGP_ALLCSPC 2-12	SGP_NWRHIT@ 2-10, 2-16
SGP_ALLOC@ 2-12	SGP_NWRTPSC 2-10, 2-16
SGP_BESCOLT 2-12, 2-15	SGP_PENDTIM 2-11, 2-12, 2-13, 2-16
SGP_BESFREE 2-12, 2-15	SGP_RDHIT@ 2-10, 2-11, 2-12, 2-13, 2-16
SGP_BESTOTL 2-12, 2-15	SGP_RDSPRSC 2-10, 2-16
SGP_BESUNCL 2-12, 2-15	SGP_READ@ 2-10, 2-11, 2-12, 2-13, 2-16

SGP_RESERV@ 2-13, 2-16
 SGP_RESPTIM 2-11, 2-12, 2-13, 2-16
 SGP_RSFNAM 2-12, 2-16
 SGP_SERVTIM 2-11, 2-12, 2-13
 SGP_SRDHIT@ 2-10, 2-13, 2-17
 SGP_SRDPRSC 2-10, 2-13, 2-17
 SGP_SWRHIT@ 2-10, 2-17
 SGP_SWRPRSC 2-10, 2-17
 SGP_TOTSPAC 2-12
 SGP_WRHIT@ 2-10, 2-11, 2-12, 2-13, 2-17
 SGP_WRITE@ 2-10, 2-11, 2-12, 2-13, 2-17
 SGP_WRPRSEC 2-10, 2-17
 SGREADNTVLn 3-7
 SGRETRYLIMn 3-8
 SGSPACHLDRn 3-8
 SGSUBTASKSn 3-8
 SGWRITNTVLn 3-8
 SIZE 2-21, 2-29, 2-33
 skeleton processing 6-28
 skeleton statements
 ISPF JCL skeleton facility 6-17
 SMF 8-3
 parameters 4-2
 SMF record types 8-3
 SMHSMUTL 2-36
 DFHSM control data set batch utility 2-36
 SMMSYSxx 2-2
 software categories supported 8-4
 space data collector
 data definition 3-8
 security 3-11
 using multiple collectors 3-10
 START_SGP 3-8, 4-6
 statements, job control 6-6
 STEP 2-33
 STEPACCTn 2-33
 STEPLIB 2-42
 Storage Management Subsystem
 data facility product 8-3
 hierarchical storage management 8-3
 removable media manager 8-3
 StorageGUARD
 copy/merge utility 3-3
 global parameters 3-6
 historical performance data collector 4-2
 SET statements 3-6
 SMF and CMF/RMF parameters 4-2
 space data collector 3-6
 data definition requirements 3-8
 security 3-11
 using multiple collectors 3-10
 StorageTeK
 HSC 8-3
 StorageTek
 VTCS 8-3
 STORCLAS 2-11, 2-13
 STORGRP 2-13
 submission, job 6-14
 substrings
 use in NEWAPPL 2-7
 symbolic substitution
 use of 2-7
 syntax 6-24
 BUDDSN command
 data sets on volume 5-20
 group of data sets 5-19
 BUDGET command
 add/change multiple application codes
 5-33
 add/change single application code 5-26,
 5-32
 field to field compare 5-24, 5-34
 field to value compare 5-23, 5-33
 list single application code 5-22
 listing multiple application codes 5-23,
 5-24
 SGCMAINT
 COMPARE statement B-20
 control statements B-3
 file control statement B-4
 option control statement B-5
 REPORT statement B-17
 RESYNC statement B-13
 VOLUME Selection statement B-14
 SYSID 2-33
 SYSIN 2-42, 6-21
 sysplex
 environments with shared DASD
 considerations 6-8
 system exits
 add exit A-9
 check exit A-10
 select exit A-13
 System Management Facility 8-3
 system parameters 6-8, 8-6

T

Tape Management software
 CA-1 8-4
 CONTROL-T 8-4
 DFSMSrmm 8-4
TAPE_CA1DSN 8-6
TAPE_CAT 8-6
TAPE_CCTLTH 8-6
TAPE_CHLQ 8-6
TAPE_CPRI 8-6
TAPE_CSEC 8-6
TAPE_CTLTRL 8-6
TAPE_CVOL 8-6
TAPE_HSCCDS 8-6
TAPE_RMMDSN 8-6
TAPE_SMFDSn 8-6
text operand
 NEWAPPL statement 2-6
THSM parameter
 BUDGET command 5-28
TODB parameter
 COMPARE statement B-20
TOTAL parameter
 BUDGET command 5-23
trend reporting
 analysis report 5-10
TSCAN command 8-5
TSO 2-40, 2-42, A-19
 BUDGET 5-7
TSO BUDGET Command
 DELETE parm 5-29
 NODELETE parm 5-29
TSO command
 reviewing an application 5-8
 updating an application 5-8
TSO commands 5-18
 Allocating SGADB 5-18
 BUDDSN command 5-18
 BUDGET command 5-21
TTEMP parameter
 BUDGET command 5-27
TVSAM parameter
 BUDGET command 5-28

U

USER 2-21, 2-29, 2-33
user fields
 use of 5-6
USERF1 parameter
 BUDGET command 5-26
USERF2 parameter
 BUDGET command 5-26
USERF3 parameter
 BUDGET command 5-27
USEVOL 2-32, 2-33
utilities
 control data set cleanup 2-36
 SMHSMUTL - DFHSM control data set
 batch 2-36

V

value subparameter
 BUDGET command 5-23
views
 online 6-14
Virtual Tape Control System 8-3
VOL 2-11, 2-13, 2-17, 2-21, 2-29, 2-33
VOLSER parameter
 BUDDSN command 5-20
VOLUME parameter
 file control statement B-4
VOLUME selection statement B-14
 EXCLUDE parameter B-14
 INCLUDE parameter B-14
VTCS 8-3, 8-4
VTOC Scan extract file 7-8

W

WARN parameter
 BUDGET command 5-28
WARNING parameter
 BUDGET command 5-28

X

XMODE 2-21, 2-29, 2-33

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. THIS AGREEMENT WILL APPLY TO THE PRODUCT, UNLESS (1) YOU AGREED TO A WEB BASED LICENSE AGREEMENT WITH BMC WHEN ORDERING THE PRODUCT, IN WHICH CASE THAT WEB BASED LICENSE AGREEMENT GOVERNS THE USE OF THE PRODUCT, OR (2) IF YOU DID NOT AGREE TO A WEB BASED LICENSE AGREEMENT WITH BMC WHEN ORDERING THE PRODUCT AND YOU HAVE A WRITTEN LICENSE AGREEMENT WITH BMC, THEN THAT WRITTEN AGREEMENT GOVERNS THE USE OF THE PRODUCT. THE ELECTRONIC AGREEMENT PROVIDED WITH THE PRODUCT AS PART OF THE INSTALLATION OF THE PRODUCT WILL NOT APPLY. 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 by 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. BMC 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 BMC Products are and contain valuable confidential information of BMC ("Confidential Information"). Confidential Information means non-public technical and non-technical information relating to the BMC 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, audit 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

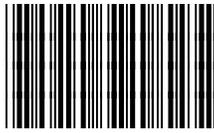
PROPRIETARY RIGHTS. ALL ARBITRATION PROCEEDINGS SHALL BE CONDUCTED IN CONFIDENCE, AND THE PARTY PREVAILING IN ARBITRATION SHALL BE ENTITLED TO RECOVER ITS REASONABLE ATTORNEYS' FEES AND NECESSARY COSTS INCURRED RELATED THERETO FROM THE OTHER PARTY.

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 EULA Int 030102

Notes



25764