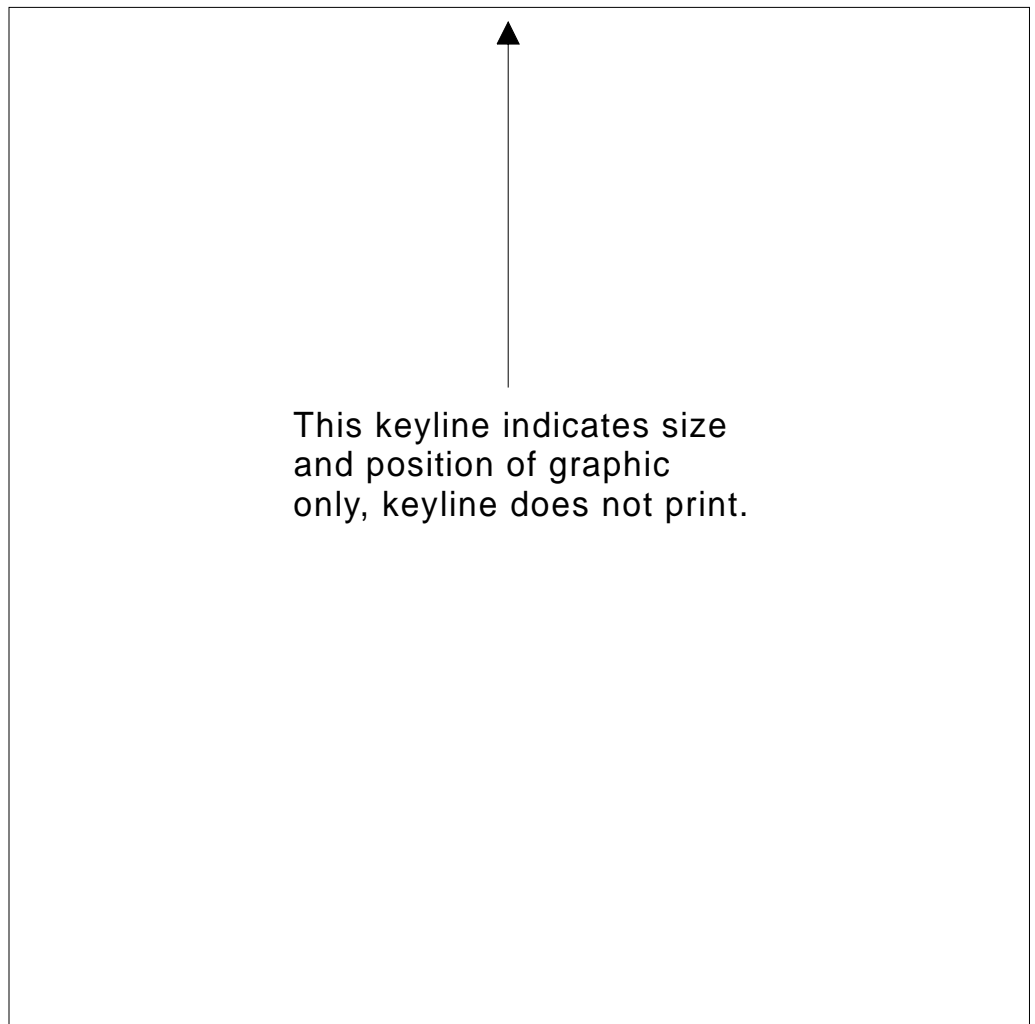


Managing Storage Groups





MVS/ESA Storage Management Library

SC26-3125-02

Managing Storage Groups

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page vii.

Third Edition (June 1994)

This edition replaces and makes obsolete the previous Edition, SC26-3125-01.

The technical changes for this edition are summarized under "Summary of Changes," and are indicated by a vertical bar to the left of a change.

This edition applies to Version 1 Release 2 of DFSMS/MVS, Program Number 5695-DF1, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

International Business Machines Corporation
Department M29, Building 86
5600 Cottle Road
SAN JOSE, CA 95193-0000
U.S.A

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1986, 1994. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	vii
Programming Interface Information	vii
Softcopy Agreement	viii
Trademarks	ix
Summary of Changes	xi
Third Edition, June 1994	xi
Product Highlights, Version 1 Release 2	xi
Publication Updates, Version 1 Release 2	xiv
Second Edition, March 1993	xiv
Product Highlights, Version 1 Release 1	xiv
First Edition, July 1992	xvii
Preface	xix
Required Publications	xix
The Storage Management Library	xix
Related Publications	xx
Referenced Publications	xxi
Chapter 1. Introducing System-Managed Storage	1
The DFSMS Environment for MVS/ESA	1
Benefits of System-Managed Storage	2
Managing Data with SMS	4
Using SMS Classes and Groups	4
Using Aggregate Groups	11
Using Automatic Class Selection Routines	11
Defining the Storage Management Subsystem Configuration	13
Coexistence Considerations	14
Sharing System-Managed and Non-System-Managed Data	14
Sharing Catalogs in Multisystem Environments	14
VTOC Compatibility Between MVS/XA and MVS/ESA Systems	16
Sharing DASD Between MVS/ESA and MVS/XA Systems	16
Sharing DASD Between MVS/ESA Systems	17
Using System Group Name with Pre-DFSMS/MVS 1.2.0 Systems	18
Using Different Releases of DFDSS, DFHSM, and DFSMS/MVS Programs	19
Hardware Considerations	19
Chapter 2. Planning DASD Storage Groups	21
Relating Your Storage Structure to Storage Groups	21
Making a Smooth Transition to Storage Groups	21
Understanding DASD Storage Group Types	22
Understanding DASD Storage Group Categories	23
Relating Non-Pooled Volumes to Storage Groups	27
Relating Volume Pools to Storage Groups	29
Determining DASD Storage Group Requirements	31
Determining the Number of Storage Groups You Need	31
DASD Volume Naming Conventions	33
Storage Group Naming Conventions	34
Sharing DASD Volumes and Storage Groups	35
Planning for the Number of Volumes	36

Configuring Pool Storage Groups for Performance and Availability	37
Using Storage Class to Define Performance and Availability Goals	37
Using Cache in the DFSMS Environment	38
Using Striping to Improve Sequential Processing Performance	40
Configuring for Data Availability	41
Chapter 3. Implementing DASD Storage Groups	43
Preparing the Volumes for Storage Groups	43
Defining the Hardware Configuration	43
Adding New Volumes	43
Converting Existing Volumes	46
Modifying the Volume Attribute List	47
Removing Volumes	48
Defining a Pool Storage Group	48
Choosing Automatic Storage Group Processing Options	49
Using Guaranteed Backup Frequency	49
Setting Thresholds for Storage Groups	50
Specifying Storage Group and Volume Connectivity	54
Using Storage Overflow to Prevent Allocation Failures	54
Managing Data with Pool Storage Groups	56
Sample Pool Storage Groups	56
Writing the Storage Group ACS Routine for DASD Data	58
Writing a Storage Class ACS Routine for Temporary Data Sets	59
Chapter 4. Maintaining Pool Storage Groups	63
Adding or Deleting Storage Groups or Devices	63
Monitoring Pool Storage Groups	64
Using the DCOLLECT Command	64
Using SLR and RMF Reports to Verify Performance	68
Managing Availability for Pool Storage Groups	69
Backing up and Recovering DASD Volumes	70
Using DFSMSHsm for Automatic Incremental Backup	70
Using DFSMSHsm for Volume Recovery	71
Using DFSMSdss for Backup and Volume Recovery	73
Using DB2 Utilities for Backup and Recovery	73
Backup and Recovery Strategies	73
Managing Space in Storage Groups	74
Managing Space with DFSMSHsm	75
Managing Space with ISMF	78
Consolidating Space on Fragmented Volumes with DFSMSdss	79
Using Database Utilities to Manage Space	79
Chapter 5. Planning and Implementing Tape Storage Groups	81
Managing Tape in the DFSMS Environment	81
Managing Tape Mounts	82
Understanding Tape Mount Management	82
Analyzing Tape Mounts	83
Defining Storage Groups for Tape Buffers	83
Writing the Storage Group ACS Routine for Tape Buffers	85
Using the Improved Data Recording Capability with DFSMSHsm	86
Using Tape Libraries	87
Advantages of Using Tape Libraries	87
Defining the Tape Configuration Database	88
Defining Tape Libraries	89

Defining Tape Storage Groups	92
Adding Tape Volumes to a Tape Library	94
Using Tape Cartridges with Different Capacities	95
Tape Volume Naming Conventions	95
Cataloging Tape Data Sets	96
Writing the Tape Storage Group ACS Routine	96
Managing Your Tape Inventory	98
Chapter 6. Planning and Implementing Object Storage Groups	99
Managing Space with OAM	99
How Object Data is Stored	99
Determining Storage Group Requirements for Object Data	100
Storage Group Naming Conventions and Volume Serial Numbers	100
Object Storage Group Types and Attributes	101
Object and Object Backup Storage Group Connectivity	101
Sample Storage Group Definitions for Objects	102
Optical Library	102
Optical Disk Drive	103
Tape Library	103
Storage Class	103
Management Class	104
Optical Volumes	104
Optical Volumes in an Optical Library	105
Adding or Deleting Objects	105
Defining the Optical Configuration Database	105
Writing the Storage Group ACS Routine for Objects	106
Backup and Recovery Strategies	109
Using OAM for Backup and Recovery	109
Using DB2 Utilities for Backup and Recovery	109
Abbreviations	111
Glossary	113
Index	121

Notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used. Any functionally equivalent program which does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, 208 Harbor Drive, Stamford, CN 06904, U.S.A.

Programming Interface Information

This book is intended to help you to do implementation of system-managed storage.

This book also documents General-Use Programming Interface and Associated Guidance Information and Product-Sensitive Programming Interface and Associated Guidance Information.

General-Use programming interfaces allow the customer to write programs that obtain the services of DFSMS/MVS.

General-Use Programming Interface and Associated Guidance Information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

_____ General-Use Programming Interface _____

General-Use Programming Interface and Associated Guidance Information...

_____ End of General-Use Programming Interface _____

Product-Sensitive programming interfaces allow the customer installation to perform tasks such as diagnosing, modifying, monitoring, repairing, tailoring, or tuning of this IBM software product. Use of such interfaces creates dependencies on the detailed design or implementation of the IBM software product. Product-Sensitive programming interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Product-Sensitive Programming Interface and Associated Guidance Information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

Product-Sensitive Programming Interface

Product-Sensitive Programming Interface and Associated Guidance Information...

End of Product-Sensitive Programming Interface

Softcopy Agreement

For online versions of this book, we authorize you to:

1. Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy; and
2. Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the Program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine readable documentation.

Trademarks

The following terms, denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation in the United States or other countries:

ACF/VTAM	ADSTAR
AIX	AIXwindows
BookManager	CICS
CICS/ESA	DB2
DATABASE 2	DFSMS
DFSMS/MVS	DFSMS/VM
DFSMSdfp	DFSMSdss
DFSMSshsm	DFSMSrmm
ECKD	Enterprise Systems Architecture/390
Enterprise Systems Architecture/370	ES/3090
ES/9000	ESA/370
ESA/390	ESCON
GDDM	Hardware Configuration Definition
Hiperbatch	Hipersorting
Hiperspace	IBM
ImagePlus	IMS/ESA
MVS/DFP	MVS/ESA
MVS/SP	MVS/XA
NetView	OpenEdition
OS/2	PR/SM
PS/2	QMF
RACF	SAA
S/390	3090
VSE/ESA	VTAM

The following terms, denoted by a double asterisk (**) in this publication, are trademarks of other companies:

CADAM	CADAM, Inc.
UNIX	X/Open Company, Ltd.

Summary of Changes

The summary of changes for DFSMS/MVS provides you with a highlight of the release as well as specific changes for this book.

Third Edition, June 1994

Product Highlights, Version 1 Release 2

The following section highlights the new enhancements that are part of DFSMS/MVS Version 1 Release 2.

Programming Support

Backup-while-open: Provides the capability of backing up CICS VSAM File Control data sets under the Storage Management Subsystem (SMS) while they are open for update. Improvements include data security and integrity enhancements for SMS-managed CICS VSAM data sets.

DFSMSShsm RECYCLE Enhancements: Enhances the DFSMSShsm RECYCLE command to improve performance and usability.

Tapes containing the lowest percentages of valid data (based on cartridge type, standard or enhanced capacity) are processed first, which provides the highest number of tapes being returned for reuse early in the recycling process.

Up to 15 recycle tape-processing tasks can run simultaneously on a processor, with three tape buffers being used for improved throughput.

Recycle processing can be set to automatically quiesce when a predefined net number of tapes are returned to scratch.

DFSMSShsm Tape I/O Improvements (Multiple Tape Buffers): Enhances the tape mount management (TMM) strategy with the following improvements to tape I/O processing:

- When the DFSMSdss data mover is used for level 0 direct-to-tape migration and backup operations, multiple output buffers are obtained in extended private storage (above the 16-megabyte line) and the number of in-storage moves are reduced when software compaction is not used.

DFSMSrmm Enhancements: DFSMSrmm provides this new support:

- Support for the IBM 3495-M10 Tape Library Dataserver
- Support for the new OAM CBRUXVNL installation exit
- Support for 4-digit device numbers
- Support for OAM object tape
- A new DFSMSrmm installation exit to provide support for returning volumes to scratch status
- New bypass label processing support
- Refresh of installation exits with DFSMSrmm active

- If DFSMSrmm is not licensed for use, the DFSMSrmm supplied version of the OAM installation exits are executed once only
- DFSMSrmm installation exits are no longer automatically disabled on failure

Distributed FileManager/MVS: Allows users from remote systems to access data on an MVS/ESA system. DFM provides Inplace access to remote files from applications running in client systems; substantially reducing the need for Upload/Download scenarios. DFSMSdss saves Distributed Data Management (DDM) attributes associated with a data set and preserves those attributes during copy, move, backup, and restore operations.

Expiration Date Override: Provides the capability of overriding the expiration date when deleting an SMS-managed DASD data set—via methods other than management class—using JCL DD statements, IEHPROGM SCRATCH, or SVC 99. While management class is still the recommended method for deleting SMS-managed data sets, the expiration date override capability gives the user the ability to delete a data set without requiring the operator to authorize the deletion.

OAM Object Tape Support: Tape is now a full-fledged member of the OAM storage hierarchy. Objects can have their primary and/or backup copy stored on tape cartridges. Objects can transition up and down the hierarchy to and from tape cartridges. IBM cartridge system tapes can be inside of or outside of an automated or manual tape library (3495 ATL/MTL, 3494).

OAM RMF Statistics: Collects transaction information from OAM to generate performance-related statistical reports. The reports can be used to monitor and improve OAM's performance.

OpenEdition MVS: Creates a hierarchical file system within a new kind of MVS data set called a hierarchical file system data set (an HFS data set). The major feature consists of the ability to define multiple directories in an HFS data set and byte access of the data files. Additional constructs called symbolic links, FIFO files, and character special files, are also supported. HFS files are accessible through the POSIX-compliant file system calls of OpenEdition MVS.

Sequential Access Method Compression: Gives users of extended sequential data sets a transparent data compression capability. This support improves DASD space utilization; improves cache utilization; reduces I/O needed to reference data; reduces channel, controller, and device busy time necessary to process a given amount of data; and exploits the new hardware compression facility, if it is available. If it is not available, a software equivalent (available with the appropriate level of MVS/SP) is used.

Enhanced Volume Selection for the Storage Management Subsystem: This support allows you to mix devices of varying performance characteristics, (that have the same device geometry), within one pool storage group. SMS selects the volumes that more closely match a user's requested performance and accessibility characteristics (as specified in the storage class).

System Group Name Support: Eliminates the eight system limitation in MVS/SP JES2 environments. Although an SMS configuration for an SMS complex can still only contain eight names, those names can now be a combination of system names and system group names. A system group is a sysplex minus any systems that are explicitly defined in the base configuration.

Four-Digit Device Numbers: This support, in conjunction with the appropriate MVS/SP level, lets you attach more devices than previously allowed. Device numbers can now be four digits, removing the previous limitation of 4 096 device numbers. The MVS/ESA architecture supports up to 65 536 device numbers. The actual number of devices attachable to your system depends on the hardware configuration of your processor and I/O devices, and the amount of virtual storage available below 16 megabytes.

Virtual Storage Constraint Relief (VSCR): Increases virtual storage available to DFSMSHsm. A greater number of tasks can be run simultaneously, and virtual storage problems relieved, by bringing the DFSMSHsm load module above the 16-megabyte line in extended private storage.

VSAM KSDS Compression: Provides VSAM users with a new extended key-sequenced data set (KSDS) that supports data compression. Data compression improves DASD space utilization; improves cache utilization; reduces I/O needed to reference data; reduces channel, controller, and device busy time necessary to process a given amount of data; and exploits the new hardware compression facility where it is available. If it is not available, a software equivalent (available with the appropriate level of MVS/SP) is used.

VSAM Partial Space Release: Provides the capability of releasing allocated but unused space in extended format VSAM KSDS data sets. It is supported with a management class parameter and JCL.

Data Class Attribute (Additional Volume Amount): ISMF has been enhanced to add a new data class attribute called ADDITIONAL VOLUME AMOUNT. This attribute specifies whether primary or secondary allocation amounts are to be used when a extended format VSAM KSDS data set extends to a new volume.

Tape Library Enhancements

3494/3495 Partitioning: Multiple Storage Management Subsystem complexes can now access a 3494/3495 without using a shared ACDS, COMMDS and TCDB. Partitioning support allows you to partition a 3494/3495 into multiple logical 3494/3495s, a partition per Storage Management Subsystem complex. The volumes cannot be shared across the partitions.

OAM Volume Not In Library Exit: This installation exit gives installations the opportunity to insert a volume into a tape library (either manual tape library daserver or automated tape library daserver) during job step setup processing, device allocation processing, or library mount processing.

Device Support

3494 Tape Library Daserver: The 3494 is an automated tape library that uses the same architecture as the 3495. A 3494 configuration can have one drive and 240 tape cartridges, up to a maximum of eight drives and 3040 cartridges. It attaches to the AS/400, RISC System/6000, and ES/9000 processor families.

3990 Model 6: The 3990 Model 6 is a new DASD control unit consisting of new electronics and licensed internal code. The 3990 Model 6 provides the following enhancements:

- The capability to establish up to 128 logical channel paths using up to 16 physical channel adapters.

- Aggregate data rate increase of up to four times that of the 3990 Model 3.
- Serviceability improvements such as the capability of synchronizing timers with the attaching hosts.

3495-M10 Tape Library Dataserver: Extends the scope of tape library support, introduced in DFSMS/MVS Version 1 Release 1, to include mountable tape cartridges in an SMS-managed manual tape library dataserver. A manual tape library dataserver is an installation-defined set of tape drives and the set of volumes that can be mounted on the drives. An SMS-managed manual tape library dataserver consists of an IBM 3495 Model M10 tape library and one to four IBM 3490 magnetic tape subsystems.

3995 Models 113 and 133: Support has been added to the OAM component of DFSMS/MVS for the IBM 3995 Models 133 and 113 optical library dataservers. The Model 133 is a channel-attached optical disk library with multi-function optical disk drives (WORM and read and write). The Model 113 is an expansion unit that attaches directly to the Model 133.

Publication Updates, Version 1 Release 2

Following are the changes that are specific to this book.

- Technical changes to reflect new features and functions
- Editorial changes to improve readability and retrievability

Second Edition, March 1993

The following section describes the highlights of DFSMS/MVS Version 1 Release 1.

Product Highlights, Version 1 Release 1

DFSMS/MVS Version 1 Release 1 integrates and expands the functions previously available in MVS/DFP Version 3 (5665-XA3), the Data Facility Hierarchical Storage Manager (DFHSM) Version 2 (5665-329), and Data Facility Data Set Services (DFDSS) Version 2 (5665-327). The functions of these previous offerings and major new functions are contained in the following four DFSMS/MVS functional components:

- DFSMSdfp—Provides storage, data, program, and device management functions
- DFSMSdss—Provides data movement, copy, backup, and space management functions
- DFSMShsm—Provides backup, recovery, migration, and space management functions
- DFSMSrmm—Provides management functions for removable media such as tape cartridges, reels, and optical volumes.

Programming Support

Removable Media Management: The DFSMSrmm functional component helps you more effectively manage the data that your business stores on removable media, including tape reels, cartridges, and other media identified to it.

DFSMSrmm provides a central, online inventory of the volumes and shelf space both on-site and at off-site storage locations. You can use DFSMSrmm to automate many of your tape management tasks.

System-managed Tape Cartridges and Tape Libraries: Tape cartridges can now be system-managed. You can define tape libraries and storage groups using ISMF. These tape libraries can contain 3490 or 3490E tape devices. The associated storage groups can contain IBM Cartridge System Tape or IBM Enhanced Capacity Cartridge System Tape. You can use the IBM 3495 Tape Library Dataserver to automate tape cartridge mounts and storage for 3490 and 3490E tape devices.

Volume Mount Analyzer: The volume mount analyzer is a new tool that uses SMF records to provide a comprehensive analysis of tape data. It also identifies suitable data set candidates for use with the tape mount management technique.

Concurrent Copy: Concurrent copy allows DFSMSdss to backup or copy data sets without locking users out of the data set until the copy is complete. The backup or copy appears as if it were taken instantaneously. You can control the use of concurrent copy using the new storage class attribute, ACCESSIBILITY, and the new management class attributes, BACKUP COPY TECHNIQUE and ABACKUP COPY TECHNIQUE. To use concurrent copy to copy data, the data must reside on a DASD volume attached through an IBM 3990 Model 3 Extended Platform Storage Control or an IBM 3990 Model 6.

Program Library Support for PDSEs: DFSMS/MVS provides the program management binder and program management loader for enhanced program management. The binder relaxes many of the limitations of the MVS/DFP linkage editor and loader, and supports a new executable unit called a *program object*. The program object eliminates many of the restrictions of the traditional load module. The loader loads both program objects and traditional load modules.

Striping Support for Sequential Data Sets: A new type of data set, extended sequential, can be defined and accessed with BSAM and QSAM. An extended sequential data set consists of stripes which reside on separate DASD volumes. When data is written to a striped data set, the data is interleaved among the stripes. A striped data set differs from a non-extended multivolume data set in that data is written to one volume of a multi-volume data set until the space for the data set on the volume is filled. All extended sequential data sets must be system-managed.

You can use striping to transfer data between DASD and memory at a faster rate than an individual DASD can handle. It is a performance enhancement mainly used for large, physical sequential data sets with high I/O activity. Whether a physical sequential data set is striped is mostly transparent to an application program. Changes have been made to data class and storage class to allow users to define extended sequential data sets. This type of data set requires the 3990-3 with Extended Platform, or 3990-6, or cached 9343 Storage Control, and ESCON channel attachment.

31-bit support for BSAM, QSAM, BPAM, and BDAM: The BDAM, BPAM, BSAM, and QSAM access methods support 31-bit addresses for most macros and allow data buffers to reside above the 16MB line for non-striped and extended sequential data sets, allowing for possible performance enhancements.

Aggregate Backup and Recovery Support (ABARS): You can use Storage Management Subsystem management classes to more easily manage your aggregate groups, or groups of data sets that you want backed up together for disaster

recovery. DFSMS/MVS management class attributes include new retention attributes for aggregate groups.

Note: Because these requirements are determined by the management class of the aggregate group, the aggregate group application does not contain the fields that MVS/DFP contained for expiration date, destination, or toleration enqueue failure. Thus, if you are migrating to DFSMS/MVS from MVS/DFP, you should either print or display and write down the values of the three deleted aggregate group fields before you IPL the system with DFSMS/MVS.

Aggregate backup and recovery support has also been enhanced to include:

- The ability to create multiple aggregate backup copies
- The ability to include the integrated catalog facility user catalog and alias definitions in the aggregate backup
- The automatic calculation of required storage for recovery
- Expanded naming conflict resolution
- A reduction in the DASD storage space required for recovery by allowing recovery to tape
- An option that deletes source user DASD data sets and removes source tape data sets from the catalog after an aggregate has been successfully backed up.

Enhanced Dynamic Cache Management: DFSMS/MVS provides an enhancement to dynamic cache management. The enhancement further reduces the need for manual cache tuning and enables as many data sets as possible to benefit from the cache and non-volatile storage. A new cache management algorithm optimizes the selection of data sets that are candidates for caching, providing improved performance when the 3990 Model 3 controller's cache becomes overloaded.

Data Set Performance Statistics: DFSMS/MVS provides new SMF type 42 subtype records statistics you can use to analyze data set performance. I/O statistics are available for system-managed data sets, providing information such as the total number of I/O operations and the average I/O response time. Additional statistics are also available for system-managed data sets residing on DASD volumes connected with a 3990 Model 3 Storage Control. These statistics include the total number of I/O operations for data sets that are candidates for caching, and the number of I/O operations that actually used cache for those data sets. All of these statistics are also available by storage class.

Sharing Partitioned Data Sets Extended (PDSEs) Among Systems:

DFSMS/MVS allows users on different MVS/ESA systems to access the same partitioned data set extended (PDSE) simultaneously. Users on multiple MVS systems can concurrently create, read, or replace members of the PDSE.

Improved Performance for Copying PDSEs: DFSMS/MVS improves the performance of copying PDSEs; making it comparable to the performance of copying partitioned data sets with IEBCOPY and DFSMSdss.

Enhanced Device Console Services: Device console services are also improved, allowing you to issue the DISPLAY SMS,OAM and DISPLAY SMS,OSMC operator commands from a TSO terminal in CONSOLE mode. Also, messages for Storage Management Subsystem operator commands that are issued by the system console operator are routed back only to the console on which the command was issued.

Improved Support for Empty Data Sets: Users will not receive I/O errors or read residual data if users attempt to read certain data sets before the data has been written. This enhancement applies to all system-managed sequential data sets, striped and non-striped, and all system-managed data sets with undefined data set organizations.

Device Support

DFSMS/MVS introduces the following new device support:

IBM 3495 Tape Library Dataserver: DFSMS/MVS supports the IBM 3495 Tape Library Dataserver, which you can use to automate your tape libraries.

IBM S/370 and S/390 Optical Media Attach/2: DFSMS/MVS supports the IBM S/370 and S/390 Optical Media Attach/2, which enables MVS users to use OS/2 to read in softcopy books or other files distributed on storage media supported by OS/2 including hard disks, CD-ROMs, and floppy disks. The IBM S/370 and S/390 Optical Media Attach/2 emulates an IBM 3422 Magnetic Tape Subsystem, and is defined to the system with the device type 3423.

DFSMS/MVS supports all models of the 3390, 9345, and 3380, including devices that emulate the 3390, 9345, or 3380. However, DFSMS/MVS Version 1 Release 1 is the last release to support 3380 standard models.

First Edition, July 1992

This edition is based on and replaces *MVS/ESA Storage Management Library: Managing Storage Pools*, SC26-4656.

Preface

This book is intended for storage administrators—those people in a data processing organization who are responsible for managing data sets and objects, storage, and the storage hardware configuration. This book describes how to plan for, design, implement, and maintain system-managed storage groups.

This book is part of the MVS/ESA Storage Management Library—a set of books that outlines efficient storage management strategies and techniques for MVS/ESA systems. Although most of the tools and techniques illustrated in this book are specific to an MVS/ESA environment, you can apply the concepts of efficient storage management to the MVS/XA and MVS/370 operating systems.

Required Publications

To understand this manual, you need to be familiar with the information in the following publications:

MVS/ESA SML: Implementing System-Managed Storage

MVS/ESA SML: Managing Data

DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference

For information on the product requirements for the DFSMS environment, see *DFSMS/MVS V1R2 General Information*, GC26-4900.

The Storage Management Library

Figure 1 shows the Storage Management Library. This library does not replace the publications that document the various IBM hardware and software products used for storage management. Instead, it provides a framework to help storage administrators identify when information in another library is relevant to a particular task.

The Storage Management Library is also available in BookManager-readable form on the *IBM Online Library Omnibus Edition MVS Collection*, SK2T-0710.

Figure 1 (Page 1 of 2). The Storage Management Library

Publication	Description
<i>MVS/ESA Storage Management Library: Leading a Storage Administration Group</i> , SC26-3126	Provides advice for the data processing manager on setting up a storage administration group, providing service to user groups, and planning for storage management in the future.
<i>MVS/ESA Storage Management Library: Implementing System-Managed Storage</i> , SC26-3123	Provides system programmers, storage administrators, and other data processing professionals with information on how to plan and implement the conversion of an existing MVS installation to system-managed storage. Former title: <i>Storage Management Subsystem Migration Planning Guide</i> .

Figure 1 (Page 2 of 2). The Storage Management Library

Publication	Description
<i>MVS/ESA Storage Management Library: Managing Data</i> , SC26-3124	Provides guidance to system programmers, storage administrators, and other data processing professionals on how to establish and enforce storage management policies; manage data sets, objects, and database data; and protect the data in a system-managed environment. Former title: <i>Managing Data Sets and Objects</i> .
<i>MVS/ESA Storage Management Library: Managing Storage Groups</i> , SC26-3125	Provides system programmers, storage administrators, and data processing professionals information on how to plan, implement, and maintain storage groups for DASD, tape, and optical volumes. Former title: <i>Managing Storage Pools</i> .
<i>IBM Online Library Omnibus Edition MVS Collection</i> , SK2T-0710	The Storage Management Library in BookManager-readable format is included on this CD-ROM.
<i>MVS/ESA Storage Management Library</i> , SBOF-3012	The complete Storage Management Library in hardcopy format.

Related Publications

The following publications contain additional information related to storage management, but are not cited in this publication.

Publication	Order Number
<i>IBM 3380 Storage Subsystem Library</i>	GBOF-1756
<i>IBM 3390 Storage Subsystem Library</i>	GBOF-3121
<i>IBM 9340 Storage Subsystem Library for MVS</i>	GBOF-3127
<i>MVS/ESA Initialization and Tuning Guide</i>	SC28-1451
<i>MVS/ESA JCL User's Guide</i>	GC28-1473
<i>MVS/ESA JES3 Initialization and Tuning Guide</i>	SC28-1455
<i>MVS/ESA JES3 Introduction</i>	GC28-1421
<i>MVS/ESA System Commands</i>	GC28-1442
<i>MVS/ESA System Data Set Definition</i>	GC28-1432
<i>Resource Access Control Facility (RACF) Command Language Reference</i>	SC28-0733
<i>Resource Access Control Facility (RACF) Security Administrator's Guide</i>	SC28-1340
<i>Using the IBM 3380 Direct Access Storage in an MVS Environment</i>	GC26-4492
<i>Using IBM 3390 Direct Access Storage in an MVS Environment</i>	GC26-4574
<i>Using IBM 9340 Direct Access Storage Subsystems in an MVS Environment</i>	GC26-4646
<i>IBM 3494 Tape Library Dataserver Introduction and Planning Guide</i>	GA32-0279
<i>IBM 3495 Tape Library Dataserver Model M10 Introduction and Planning Guide</i>	GA32-0275

Referenced Publications

Within the text, references are made to the following publications:

Short Title	Publication Title	Order Number
DB2 Administration Guide	<i>IBM DATABASE 2 Version 3 Administration Guide</i>	SC26-4888
DFSMS/MVS V1R2 Access Method Services for ICF	<i>DFSMS/MVS Version 1 Release 2 Access Method Services for the Integrated Catalog Facility</i>	SC26-4906
DFSMS/MVS V1R2 DFSMSHsm Implementation and Customization Guide	<i>DFSMS/MVS Version 1 Release 2 DFSMSHsm Implementation and Customization Guide</i>	SH21-1078
DFSMS/MVS V1R2 Managing Catalogs	<i>DFSMS/MVS Version 1 Release 2 Managing Catalogs</i>	SC26-4914
DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support	<i>DFSMS/MVS Version 1 Release 2 Object Access Method Planning, Installation, and Storage Administration Guide for Object Support</i>	SC26-4917
DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Tape Libraries	<i>DFSMS/MVS Version 1 Release 2 Object Access Method Planning, Installation, and Storage Administration Guide for Tape Libraries</i>	SC26-3051
DFSMS/MVS V1R2 Planning for Installation	<i>DFSMS/MVS Version 1 Release 2 Planning for Installation</i>	SC26-4919
DFSMS/MVS V1R2 DFSMSDss Storage Administration Guide	<i>DFSMS/MVS Version 1 Release 2 DFSMSDss Storage Administration Guide</i>	SC26-4930
DFSMS/MVS V1R2 DFSMSHsm Storage Administration Guide	<i>DFSMS/MVS Version 1 Release 2 DFSMSHsm Storage Administration Guide</i>	SH21-1076
DFSMS/MVS V1R2 DFSMSDfp Storage Administration Reference	<i>DFSMS/MVS Version 1 Release 2 DFSMSDfp Storage Administration Reference</i>	SC26-4920
DFSMS/MVS V1R2 DFSMSDss Storage Administration Reference	<i>DFSMS/MVS Version 1 Release 2 DFSMSDss Storage Administration Reference</i>	SC26-4929
DFSMS/MVS V1R2 Using Data Sets	<i>DFSMS/MVS Version 1 Release 2 Using Data Sets</i>	SC26-4922
DFSMS/MVS V1R2 DFSMSRmm Guide and Reference	<i>DFSMS/MVS Version 1 Release 2 DFSMSRmm Guide and Reference</i>	SC26-4931
DFSMS/MVS V1R2 Using ISMF	<i>DFSMS/MVS Version 1 Release 2 Using the Interactive Storage Management Facility</i>	SC26-4911
3495 Tape Library Dataserver Operator's Guide	<i>IBM 3495 Tape Library Dataserver Operator's Guide</i>	GA32-0235
ICKDSF User's Guide and Reference	<i>Device Support Facilities User's Guide and Reference</i>	GC35-0033
HCD: Planning	<i>MVS/ESA HCD: Planning</i>	GC28-1445
Planning: Global Resource Serialization	<i>MVS/ESA Planning: Global Resource Serialization</i>	GC28-1450
RMF Monitor I and II Reference and User's Guide	<i>MVS/ESA Resource Measurement Facility Version 4 Monitor I and II Reference and User's Guide</i>	LY28-1007

Short Title	Publication Title	Order Number
RMF Monitor III Reference and User's Guide	<i>MVS/ESA Resource Measurement Facility Version 4 Monitor III Reference and User's Guide</i>	LY28-1008
SLR User's Guide: Reporting Planned Availability	<i>Service Level Reporter Version 3.3 User's Guide: Reporting Planned Availability</i>	SH19-6530
MVS/ESA SPL: SMF	<i>MVS/ESA System Management Facilities (SMF)</i>	GC28-1457
Hardware Configuration Definition: User's Guide	<i>MVS/ESA Hardware Configuration Definition: User's Guide</i>	SC33-6468
3990 Planning, Installation, and Storage Administration Guide	<i>IBM 3990 Storage Control Planning, Installation, and Storage Administration Guide</i>	GA32-0100

Chapter 1. Introducing System-Managed Storage

System-managed storage is the IBM* automated approach to managing auxiliary storage. It uses software programs to manage data security, placement, migration, backup, recall, recovery, and deletion to ensure that current data is available when needed, and obsolete data is removed from storage.

System-managed storage is tailored to your needs. You define the requirements for performance, security, and availability, along with storage management policies used to automatically manage the direct access, tape, and optical devices used by its MVS/ESA* operating systems.

The combination of system-managed storage and related hardware and software products is called the *DFSMS* environment*.

The DFSMS Environment for MVS/ESA

To implement the DFSMS environment and to take advantage of all the functions available with MVS/ESA, you need to install a specific set of software products. Figure 2 shows the relationship among the IBM products that form the DFSMS environment.

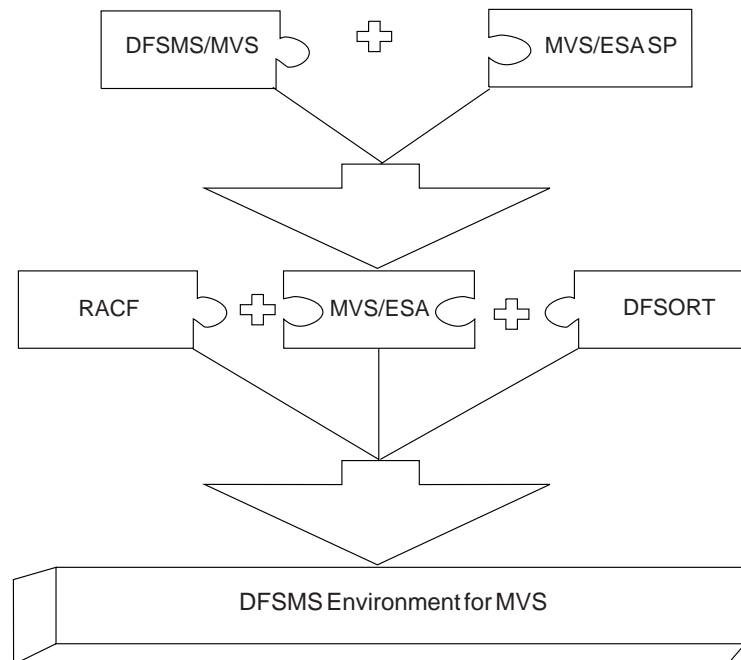


Figure 2. Products in the DFSMS Environment

The **DFSMSdfp*** functional component of DFSMS/MVS* provides the storage, program, data, and device management functions of MVS/ESA. The Storage Management Subsystem (SMS) component of DFSMSdfp is fundamental in providing these functions.

The **DFSMSdss*** functional component of DFSMS/MVS copies and moves data for MVS/ESA.

The **DFSMShsm*** functional component of DFSMS/MVS provides the backup, recovery, migration, recall, and space management functions in the DFSMS environment.

The **DFSMSrmm*** functional component of DFSMS/MVS provides the management functions for removable media, including tape cartridges and reels, and optical volumes.

DFSORT sorts, merges and copies data.

RACF* controls access to data and other resources in MVS/ESA.

Benefits of System-Managed Storage

With the Storage Management Subsystem (SMS), you can define performance goals and data availability requirements, create model data definitions for typical data sets, and automate data backup. SMS can automatically assign, based on installation policy, those services and data definition attributes to data sets when they are created. IBM storage management-related products determine data placement, manage data backup, control space usage, and provide data security.

The goals of system-managed storage are to:

- Improve the use of the storage media; for example, by reducing out-of-space abends and providing a way to set a free-space requirement.
- Reduce the labor involved in storage management by centralizing control, automating tasks, and providing interactive controls for storage administrators.
- Reduce the user's need to be concerned with the physical details of performance, space, and device management. Users can focus on using data instead of managing data.

The benefits of system-managed storage are:

Simplified Data Allocation

System-managed storage enables users to simplify their data allocations. For example, without using the Storage Management Subsystem, an MVS/ESA user would have to specify the unit and volume on which the system should allocate the data set. The user would also have to calculate the amount of space required for the data set in terms of tracks or cylinders. This means the user has to know the track size of the device which will contain the data set.

With system-managed storage, users can let the system select the specific unit and volume for the allocation. They can also specify size requirements in terms of megabytes or kilobytes. This means the user does not need to know anything about the physical characteristics of the devices in the installation.

Improved Allocation Control

System-managed storage enables you to set a requirement for free space across a set of direct access storage device (DASD) volumes. You can then provide adequate free space to avoid out-of-space abends. The system automatically places data on a volume containing adequate free space.

You can also set a threshold for scratch tape volumes in tape libraries, to ensure enough cartridges are available in the tape library for scratch mounts.

Improved Input/Output (I/O) Performance Management

System-managed storage enables you to improve DASD I/O performance across the installation and at the same time reduce the need for manual tuning by defining performance goals for each class of data. You can use cache statistics recorded in system management facilities (SMF) records to help evaluate performance. You can also improve sequential performance by using extended sequential data sets. The DFSMS environment makes the most effective use of the caching abilities of the IBM 3990 Model 3 and Model 6 Storage Controls, as well as other new models.

Automated DASD Space Management

System-managed storage enables you to automatically reclaim space which is allocated to old and unused data sets or objects. You can define policies that determine how long an unused data set or object will be allowed to reside on *primary storage* (storage devices used for your active data). You can have the system remove obsolete data by migrating the data to other DASD, tape, or optical volumes, or you can have the system delete the data. You can also release allocated but unused space which is assigned to new and active data sets.

Automated Tape Space Management

System-managed storage enables you to fully use the capacity of your tape cartridges and to automate tape mounts. Using tape mount management techniques, DFSMSHsm can fill tapes to their capacity. With 3490E tape devices, Enhanced Capacity Cartridge System Tape, 36-track recording mode, and the improved data recording capability, you can increase the amount of data that can be written on a single tape cartridge.

You can also use the IBM 3495 or 3494 Tape Library Dataserver to automatically mount tape volumes and manage the inventory in an automated tape library. If you do not have an automated tape library dataserver, you can still take advantage of system-managed tape by using manual tape libraries and the 3495 Model M10 Tape Library Dataserver.

Automated Optical Space Management

System-managed storage enables you to fully use the capacity of your optical cartridges and to automate optical mounts. Using a 3995 Optical Library Dataserver, you can automatically mount optical volumes and manage the inventory in an automated optical library.

Improved Data Availability Management

System-managed storage enables you to provide different backup requirements to data residing on the same DASD volume. Thus, you do not have to treat all data on a single volume the same way.

You can use DFSMSHsm to automatically back up CICS/ESA* and DATABASE 2* (DB2*) databases, partitioned data sets extended (PDSEs), and physical sequential, partitioned, virtual storage access method (VSAM), hierarchical file system (HFS), and direct access data sets. You can also back up other types of data and use concurrent copy to maintain access to critical data sets while they are being backed up. Concurrent Copy, along with Backup-While-Open, has an added advantage that it avoids the invalidation of a backup of a CICS VSAM KSDS due to a control area or control interval split.

You can also create a logical grouping of data sets, so that the group is backed up at the same time to allow for recovery of the application defined by the group. This is done with the aggregate backup and recovery support (ABARS) provided by DFSMSHsm.

Simplified Conversion of Data to Different Device Types

System-managed storage enables you to move data to new volumes without requiring users to update their job control language (JCL). Because users in a DFSMS environment do not need to specify the unit and volume which contains their data, it does not matter to them if their data resides on a specific volume or device type. This allows you to easily replace old devices with new ones.

You can also use system-determined block sizes to automatically reblock physical sequential and partitioned data sets that can be reblocked.

Managing Data with SMS

In the DFSMS environment, you use SMS classes and groups to set service requirements, performance goals, and data definition models for your installation. You use the Interactive Storage Management Facility (ISMF) to create the appropriate classes and groups, and Automatic Class Selection (ACS) routines to assign them to data according to your installation's policies.

Using SMS Classes and Groups

On systems that do not use DFSMS, storage management consists mostly of manual operations performed on individual data sets, and manual and automated operations performed on volumes. With SMS, you can automate storage management for individual data sets and objects, and for DASD, optical, and tape volumes. You use SMS classes and groups to define the goals and requirements that the system should meet for a data set or object. Figure 3 on page 5 shows the relationship of the classes and groups to your goals and requirements.

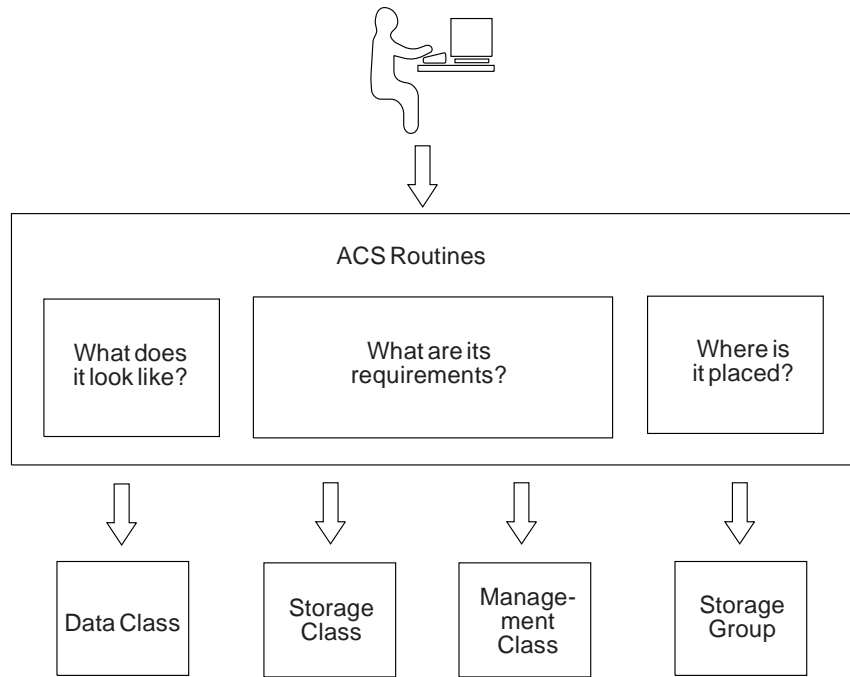


Figure 3. Allocating Data Sets or Storing Objects. You use data class to define model allocation characteristics for data sets; storage class to define performance and availability goals; management class to define backup and retention requirements; and storage group to create logical groupings of volumes to be managed as a unit.

Figure 4 shows how a data set, object, DASD volume, tape volume, or optical volume becomes system-managed.

	DASD	Optical	Tape
Data Set ₁	Assign Storage Class	Not applicable	Not system-managed ₂
Object ₃	Stored	Stored	Stored
Volume	Assign Storage Group	Define OAM Storage Groups	Assign Storage Group ₄

Figure 4. When A Data Set, Object, or Volume Becomes System-Managed

Notes:

1. A DASD data set is *system-managed* if you assign it a storage class. If you do not assign a storage class, the data set is directed to a *non-system-managed* DASD or tape volume — one that is not assigned to a storage group.
2. You can assign a storage class to a tape data set to direct it to a *system-managed* tape volume. However, only the tape volume is considered *system-managed*, not the data set.
3. OAM objects each have a storage class; therefore, objects are *system-managed*. The optical or tape volume on which the object resides is also *system-managed*.
4. Tape volumes are added to tape storage groups in tape libraries when the tape data set is created.

Using Data Classes

A *data class* is a collection of allocation and space attributes that you define. It is used when data sets are created. You can simplify data set allocation for your users by defining data classes that contain standard data set allocation attributes. You can use data classes with both system-managed and non-system-managed data sets, but some data class characteristics are only available with system-managed requests.

Data class attributes define space and data characteristics of data sets that are normally specified on JCL DD statements, TSO/E ALLOCATE commands, access method services (IDCAMS) DEFINE commands, dynamic allocation requests, and ISPF/PDF panels. For tape data sets, data class attributes can also specify the type of cartridge and recording method, and if the data is to be compacted. Users then need only specify the appropriate data classes to create standardized data sets.

You can use data class to specify that a sequential data set should be striped. Extended sequential data sets are allocated across DASD volumes to sustain a data rate you specify.

You can also use the data class automatic class selection (ACS) routine to automatically assign data classes to new data sets. For example, data sets with the low-level qualifiers LIST, LISTING, OUTLIST, or LINKLIST are usually utility output data sets with similar allocation requirements, and can all be assigned the same data class.

As Figure 5 shows, data sets can be assigned a data class during data set creation.

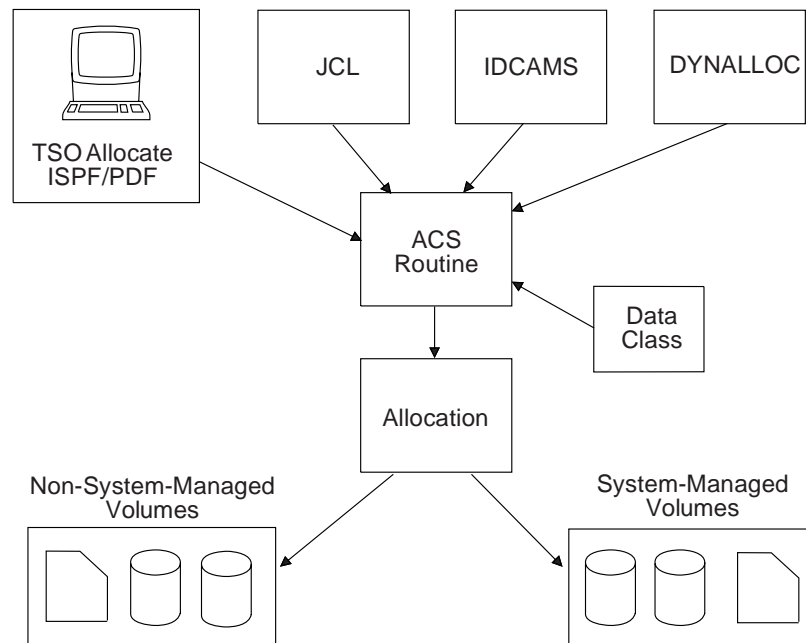


Figure 5. Using Data Class. Data classes can be used for new allocations of both system-managed and non-system-managed data sets.

Even though data class is optional, we usually recommend that you assign data classes to system-managed and non-system-managed data. Although the data class is not used after the initial allocation of a data set, the data class name is kept in the catalog entry for system-managed data sets for future reference. The data class name is not saved for non-system-managed data sets, although the allocation attributes in the data class are used to allocate the data set.

For objects on tape, we recommend that you do not assign a data class via the ACS routines. To assign a data class, specify the name of that data class on the SETOAM command.

If you change a data class definition, the changes only affect new allocations. Existing data sets allocated with the data class are not changed.

Using Storage Classes

A *storage class* is a collection of performance goals and availability requirements that you define. It is used to select a device to meet those goals and requirements. Only system-managed data sets and objects can be assigned a storage class. Storage classes free users from having to know about the physical characteristics of storage devices and manually placing their data on appropriate devices.

Some of the availability requirements you can specify with storage classes can only be met by DASD volumes attached through a 3990 Model 3 or Model 6 Storage Control or a similar device. You can specify attributes that require the use of the 3990 Model 3 or Model 6 Storage Control dual copy, DASD fast write, and concurrent copy features. The performance goals can be met through devices attached through storage controls with or without cache, depending on the goals you set. Figure 6 shows the storage control configurations needed to use all storage class attribute values.

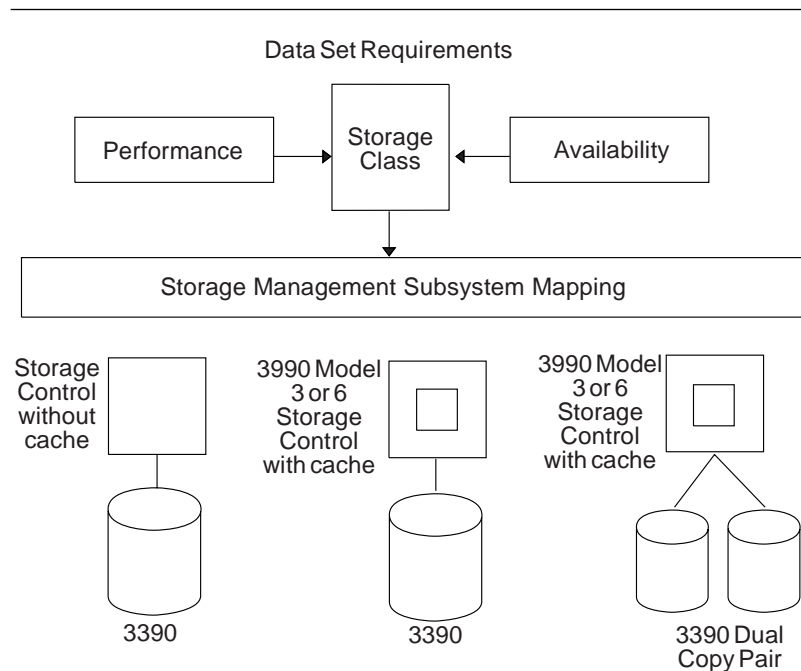


Figure 6. Using Storage Class. Storage classes make the best use of 3990 Model 3 or 6 Storage Controls.

With storage class, you can assign a data set to dual copy volumes to ensure continuous availability for the data set. With dual copy, two current copies of the data set are kept on separate DASD volumes. If the volume containing the primary copy of the data set is damaged, the companion volume is automatically brought online and the data set continues to be available and current.

You can use the ACCESSIBILITY attribute of the storage class to request that concurrent copy be used when data sets or volumes are backed up.

You can specify an I/O response time objective with storage class. During data allocation, the system attempts to select the closest available volume to the specified performance objective.

For objects, the system uses the performance goals you set in the storage class to place the object on DASD, optical, or tape volumes. The storage class is assigned to an object when it is stored or when the object is transitioned. The ACS routines can override this assignment.

If you change a storage class definition, the changes affect the performance service levels of existing data sets that are assigned that class when the data sets are subsequently opened. However, the definition changes do not affect the location or allocation characteristics of existing data sets.

Using Management Classes

A *management class* is a collection of management attributes that you define. It is used to control the retention, migration, backup and release of allocated but unused space for data sets, or to control the retention, backup, and class transition of objects. Management classes let you define management requirements for individual data sets, rather than defining the requirements for entire volumes.

If you do not explicitly assign a management class to a system-managed data set or object, the system uses the default management class. You can define your own default management class when you define your SMS base configuration.

For objects, you can:

- Assign a management class when it is stored, or
- Assign a new management class when the object is transitioned, or
- Change the management class by using the OAM Application Programming Interface (OSREQ CHANGE function).

The ACS routines can override this assignment for objects.

Figure 7 on page 9 shows that you can use management class attributes to do the following:

- Allow early migration for old generations of a generation data group (GDG) by specifying the maximum number of generations to be kept on primary storage, and what to do with rolled-off generation data sets
- Delete selected old and unused data sets from DASD volumes
- Release allocated but unused space from data sets
- Migrate unused data sets to tape or DASD volumes
- Specify how often to back up data sets, and whether concurrent copy should be used during backup

- Specify how many backup versions to keep for data sets
- Specify how long to save backup versions
- Specify the number of versions of aggregate backups to keep and how long to retain those versions
- Establish the expiration date for objects
- Establish transition criteria for objects
- Indicate if automatic backup is needed for objects

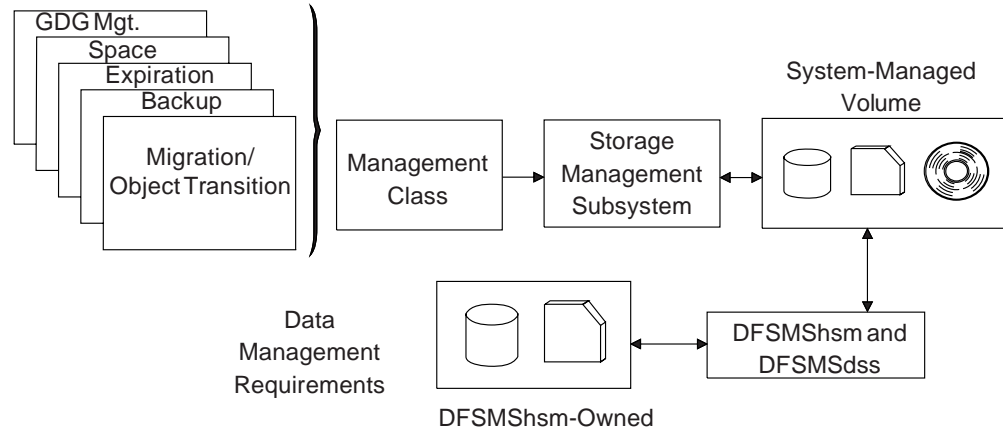


Figure 7. Using Management Class

By classifying your data according to management requirements, you can define unique management classes to fully automate your data set and object management. For example, you can use management classes to control the migration of CICS* user databases, DB2 user databases and archive logs, test systems and their associated data sets, and IMS archive logs. You can specify that DB2 image copies, and IMS image copies and change accumulation logs, be written to primary volumes and then migrated directly to migration level 2 tape volumes.

For objects, you use the class transition attributes to define when an object is eligible for a change in its performance objectives or management characteristics. For example, after a certain number of days you might want to move an object from a high-performance DASD volume to a slower optical volume. You can also use the management class to specify that the object should have a backup copy made when the OAM Storage Management Component (OSMC) is executing.

If you change a management class definition, the changes affect the management requirements of existing data sets and objects that are assigned that class.

Using Storage Groups

A *storage group* is a collection of storage volumes and attributes that you define.

The collection can be a group of:

- System paging volumes
- DASD volumes
- Tape volumes
- Optical volumes
- Combination of DASD and optical volumes that look alike

- DASD, tape and optical volumes treated as a single object storage hierarchy.

Storage groups, along with storage classes, help reduce the requirement for users to understand the physical characteristics of the storage devices which contain their data.

You can direct new data sets to as many as 15 storage groups, although only one storage group is selected for the allocation. The system uses the storage class attributes, volume and storage group SMS status, MVS volume status, and available free space to determine the volume selected for the allocation. In a tape environment, you can also use tape storage groups to direct a new tape data set to an automated or manual tape library.

DFSMSHsm uses some of the storage group attributes to determine if the volumes in the storage group are eligible for automatic space or availability management.

Figure 8 is an example of using storage groups to group storage volumes for specific purposes.

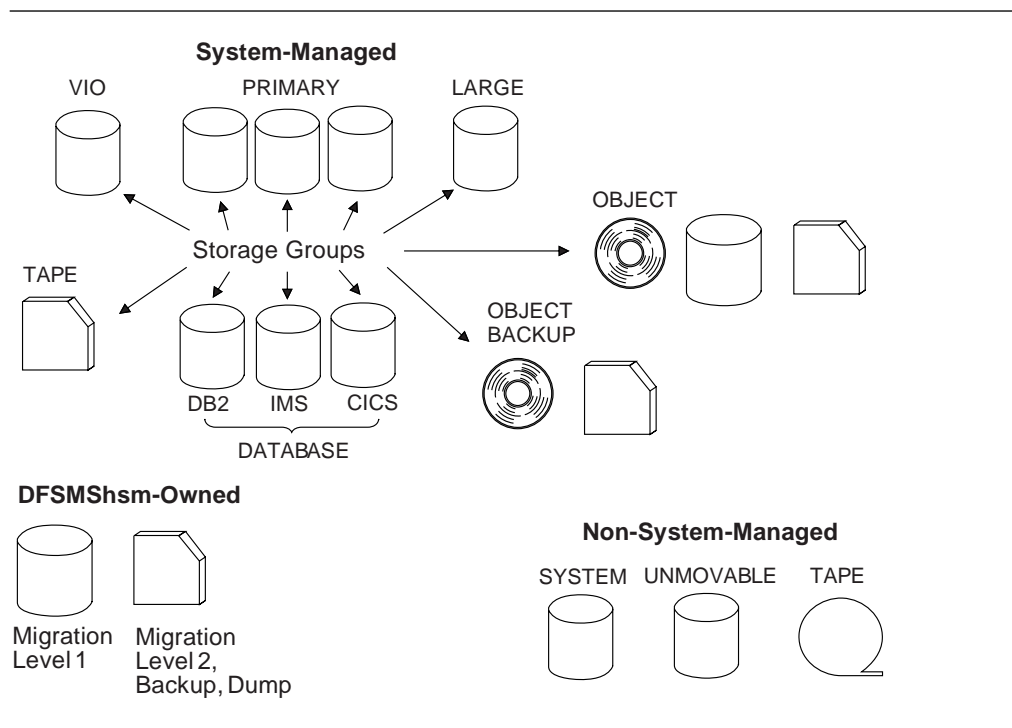


Figure 8. Using Storage Groups. In this example, DASD volumes are grouped so that primary data sets, large data sets, DB2 data, IMS data, and CICS data are all separated. The VIO storage group uses system paging volumes for small temporary data sets. The tape storage groups are used to group tape volumes that are held in tape libraries. The object storage group can span optical, DASD and tape volumes; the object backup storage group can contain either optical or tape volumes within one OAM invocation. Some volumes are not system-managed, and other volumes are owned by DFSMSHsm for use in data backup and migration. DFSMSHsm migration level 2 tape cartridges can be system-managed if you assign them to a tape storage group.

Unlike data, storage, and management classes, users cannot specify a storage group when they allocate a data set, although they can specify a unit and volume. Whether or not you honor their unit and volume request is your decision, but we recommend you discourage users from directly requesting specific devices. It is more effective for your users to specify the logical storage requirements of their

data by storage and management class, which you can then verify in the automatic class selection routines.

For objects, there are two types of storage groups, OBJECT and OBJECT BACKUP. An OBJECT storage group is assigned by OAM when the object is stored; the storage group ACS routine can override this assignment. There is only one OBJECT BACKUP storage group and all backup copies of all objects are assigned to this storage group.

Using Aggregate Groups

An *aggregate group* is a collection of related data sets and control information that have been pooled to meet a defined backup or recovery strategy. If a disaster occurs, you can use these backups at a remote or local site to recover critical applications.

You can use aggregate groups as a supplement to using management class for applications that are critical to your business. You can associate an aggregate group with a management class. The management class specifies backup attributes for the aggregate group, such as the copy technique for backing up DASD data sets on primary volumes, the number of aggregate versions to retain, and how long to retain versions. Aggregate groups simplify the control of backup and recovery of critical data sets and applications.

Although SMS must be used on the system where the backups are performed, you can recover aggregate groups to systems that are not using SMS, provided that the groups do not contain data which requires that SMS be active, such as PDSEs. You can use aggregate groups to transfer applications to other data processing installations or migrate applications to newly-installed DASD volumes. You can transfer the application's migrated data, along with its active data, without recalling the migrated data.

Using Automatic Class Selection Routines

General-use programming interface

You use automatic class selection (ACS) routines to assign class and storage group definitions to data sets, database data, and objects. You write ACS routines using the ACS language, which is a high-level programming language. Once written, you use the ACS translator to translate the routines to object form so they can be stored in the SMS configuration.

The ACS language contains a number of read-only variables, which you can use to analyze new data allocations. For example, you can use the read-only variable &DSN to make class and group assignments based on data set or object collection name, or &LLQ to make assignments based on the low-level qualifier of the data set or object collection name. You cannot alter the value of read-only variables.

You use the four read-write variables to assign the class or storage group you determine for the data set or object, based on the routine you are writing. For example, you use the &STORCLAS variable to assign a storage class to a data set or object.

For a detailed description of the ACS language and its variables, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

For each SMS configuration, you can write as many as four routines: one each for data class, storage class, management class, and storage group. Use ISMF to create, translate, validate and test the routines.

Figure 9 shows the order in which ACS routines are processed. Data can become system-managed if the storage class routine assigns a storage class to the data, or if it allows a user-specified storage class to be assigned to the data. If this routine does not assign a storage class to the data, the data cannot reside on a system-managed volume.

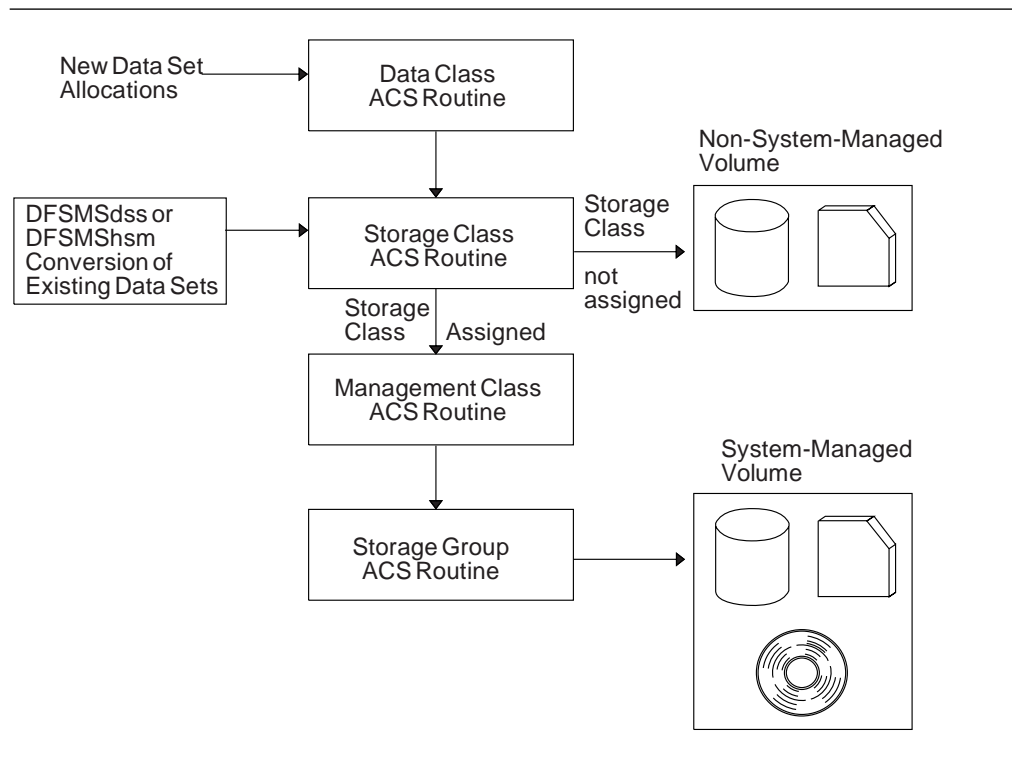


Figure 9. Processing ACS Routines

Because data allocations, whether dynamic or through JCL, are processed through ACS routines, you can enforce installation standards for data allocation on system-managed and non-system-managed volumes. ACS routines also enable you to override user specifications for data, storage, and management class, and requests for specific storage volumes.

You can use the ACS routines to determine the SMS classes for data sets created by the Distributed FileManager/MVS. If a remote user does not specify a storage class, and if the ACS routines decide that the data set should not be system-managed, the Distributed FileManager/MVS terminates the creation process immediately and returns an error reply message to the source. Therefore, when you construct your ACS routines, consider the potential data set creation requests of remote users.

End of General-use programming interface

Defining the Storage Management Subsystem Configuration

An SMS configuration is composed of a set of data class, management class, storage class, storage group, optical library and drive definitions, tape library definitions, and ACS routines to assign the classes and groups. It also includes the aggregate group definitions and the SMS base configuration. The SMS base configuration contains default information such as default management class and default device geometry. It also identifies the systems in the installation for which the subsystem manages storage.

This information is stored in SMS *control data sets*, which are VSAM linear data sets. You can define these control data sets using the access method services DEFINE CLUSTER command.

You must define the control data sets before activating SMS. Although you need only allocate the data sets from one system, the active control data set (ACDS) and communications data set (COMMDS) must reside on a device that can be accessed by every system to be managed with the SMS configuration.

SMS uses the following types of control data sets:

Source Control Data Set (SCDS)

This contains the information that defines a single storage management policy, called an SMS configuration. You can define more than one SCDS, but only one can be used to activate a configuration at any given time.

Active Control Data Set (ACDS)

This contains an SCDS that has been activated to control the storage management policy for the installation. When you activate a configuration, SMS copies the existing configuration from the specified SCDS into the ACDS. By using copies of the SMS classes, groups, volumes, optical libraries, optical drives, tape libraries, and ACS routines rather than the originals, you can change the current storage management policy without disrupting it. For example, while SMS uses the ACDS, you can:

- Create a copy of the ACDS
- Create a backup copy of an SCDS
- Modify an SCDS
- Define a new SCDS

The ACDS must reside on a shared device to ensure that all systems in the installation use the same active configuration.

Communications Data Set (COMMDS)

Contains the name of the ACDS and enables communication between SMS systems in a multisystem environment. The COMMDS also contains space statistics, SMS status, and MVS status for each system-managed volume.

Although only one COMMDS is used at a time for an SMS installation, we recommend that you have more COMMDSs on different volumes for recovery purposes.

Coexistence Considerations

This section discusses coexistence for environments using any combination of DFSMS/MVS, MVS/DFP Version 3, and MVS/XA* Data Facility Product Version 2.

Sharing System-Managed and Non-System-Managed Data

You can share system-managed data sets between systems that are using SMS and share the same control data sets. You can share any non-system-managed data sets between MVS/ESA and MVS/XA operating systems without regard for SMS or class definitions. However, you must ensure that operating systems using SMS do not share system-managed data with MVS/XA Data Facility Product Version 2 operating systems.

If you transfer data between operating systems using the TSO TRANSMIT and RECEIVE commands, or the NetView* File Transfer Program Version 2 for MVS, SMS class names are not transmitted with the data set because SMS control information is stored externally, not in the data set.

If SMS receives a data set from either a system-managed or non-system-managed environment, and ACS processing determines that the data set is to be system-managed, SMS assigns classes as for any new data set.

If the data set was originally system-managed and is transmitted to an MVS/XA Data Facility Product Version 2 operating system, the target system sees no indication of system management, and allocation proceeds normally.

If you are sending application JCL to other sites that use the facilities provided by DFSMS/MVS or MVS/DFP Version 3 (such as JCL parameters used with SMS), you need to synchronize the use of SMS at the sites.

Sharing Catalogs in Multisystem Environments

To successfully share a catalog between an MVS/XA Data Facility Product Version 2 operating system and an MVS/ESA operating system, you must ensure that all of the following are true:

- The catalog does not reside on a system-managed volume.
- No data sets cataloged in the catalog reside on a system-managed volume.
- The appropriate toleration PTFs have been applied to the Version 2 system.
- The volume is initialized as shared.
- The catalog is defined with SHAREOPTIONS(3 4).

In a DFSMS environment, all permanent DASD data sets must be cataloged. Only integrated catalog facility catalogs can be used with SMS. Although a catalog contains entries for system-managed data, the catalog itself does not have to be system-managed. However, we recommend that catalogs containing entries for system-managed data sets also be system-managed. A catalog can contain entries for data sets that are system-managed, and entries for data sets that are not system-managed.

A *shared catalog* is a basic catalog structure that is used by more than one system. It must be defined with SHAREOPTIONS(3 4), and reside on a DASD volume that has been initialized as shared. All VSAM volume data sets (VVDSs) are defined as shared.

By default, catalogs are defined as shared. You can specify that a catalog is not to be shared by specifying SHAREOPTIONS (3 3). Only define a catalog as unshared if you are certain it will not be shared. Place unshared catalogs on volumes that have been initialized as unshared. Catalogs that are defined as unshared that reside on shared volumes may become damaged.

If you are using SMS in a multisystem environment, ensure that all systems sharing system-managed data are using DFSMS/MVS or MVS/DFP Version 3. If a system using MVS/XA Data Facility Product Version 2 accesses a catalog that contains entries for system-managed data sets, the catalog may be permanently damaged because the MVS/XA Data Facility Product Version 2 system cannot recognize SMS information.

To share catalogs between systems when some of the systems are running DFSMS/MVS or MVS/DFP Version 3 **without SMS active**, and the others are running MVS/XA Data Facility Product Version 2, you must apply toleration PTFs to the Version 2 systems. Otherwise, an MVS/ESA catalog which is accessed by an MVS/XA Data Facility Product Version 2 system may be permanently damaged, requiring a catalog recovery. These toleration PTFs **do not** allow the MVS/XA Data Facility Product Version 2 system to recognize SMS information, however.

For information on catalog sharing, see *DFSMS/MVS V1R2 Managing Catalogs*.

Figure 10 shows how you can combine MVS/XA and MVS/ESA operating systems.

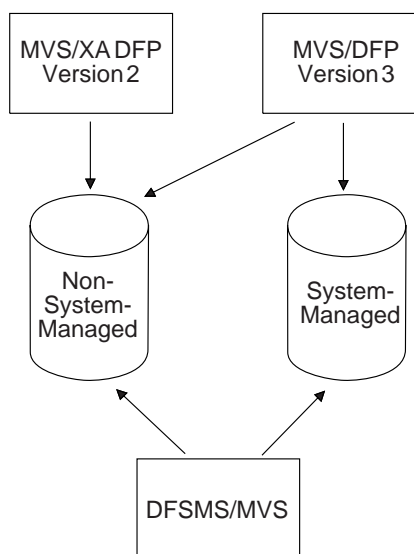


Figure 10. Configuration Combining MVS/XA and MVS/ESA Systems

If the configuration you use during SMS testing is a combination of an MVS/XA Data Facility Product Version 2 system and DFSMS/MVS or MVS/DFP Version 3, you must isolate the system-managed volumes from the MVS/XA Data Facility Product Version 2 system, preventing the MVS/XA Data Facility Product system from:

- Deleting or renaming system-managed data sets
- Creating new data sets on system-managed volumes

- Using the access method services ALTER REMOVEVOLUMES command against the master catalog to perform a VSAM volume cleanup of a system-managed volume
- Sharing catalogs with the DFSMS/MVS or MVS/DFP Version 3 system

If you have a multiaccess spool job entry subsystem (JES) installation, you might need to direct batch jobs to a particular DFSMS/MVS, MVS/DFP Version 3, or MVS/XA Data Facility Product Version 2 system to preserve the integrity of system-managed data sets.

VTOC Compatibility Between MVS/XA and MVS/ESA Systems

Product-sensitive programming interface

The structure of the volume table of contents (VTOC) is the same in MVS/XA and MVS/ESA. However, MVS/DFP Version 3 and DFSMS/MVS use a previously unused field in the format 1 data set control block (DSCB) at offset 78 (X'4E'). Before you convert from MVS/XA Data Facility Product Version 2 or prior versions or releases to MVS/DFP Version 3 or DFSMS/MVS, you should ensure that this field contains binary zeros and that there is no software on your system other than the MVS/ESA operating system that changes this field. You can get unpredictable errors if this field is used to maintain information by programs other than the operating system.

For information about inspecting the contents of DSCBs and fixing any problems that you find, see *DFSMS/MVS V1R2 Planning for Installation* and your IBM service representative.

End of Product-sensitive programming interface

Sharing DASD Between MVS/ESA and MVS/XA Systems

We recommend that you do not share DASD in an environment that has MVS/XA Data Facility Product Version 2 combined with DFSMS/MVS or MVS/DFP Version 3. If you must share DASD in such an environment, share only the non-system-managed volumes, and be sure these volumes do not contain shared catalogs.

If you attempt to share system-managed volumes between MVS/XA Data Facility Product Version 2 and DFSMS/MVS or MVS/DFP Version 3, you will have the following compatibility problems:

- The MVS/XA Data Facility Product Version 2 operating system does not track DASD space usage for SMS.
- The MVS/XA Data Facility Product Version 2 operating system does not recognize the non-VSAM volume records (NVRs) in the VSAM volume data set (VVDS). Thus:
 - Scratching or renaming a non-VSAM data set leaves NVRs in the VVDS with no associated data set.
 - VSAM volume cleanup with the ALTER REMOVEVOLUMES command removes the NVRs for non-VSAM data sets without removing the VTOC entries for those data sets.

- The MVS/XA Data Facility Product Version 2 operating system can allocate non-system-managed data sets on a system-managed volume.

Sharing DASD Between MVS/ESA Systems

You can combine up to eight MVS/ESA operating system images or system groups that are using SMS in a single installation, and share both system-managed and non-system-managed DASD volumes. The system group name represents all systems in the sysplex of the same name that are not explicitly specified in the SMS base configuration.

In an SMS complex, the operating systems communicate by sharing a common configuration stored in the ACDS and common system-managed volume statistics stored in the COMMDS. The volumes that contain these SMS control data sets must be accessible from all systems that are part of the SMS complex. The SCDS must be accessible from all systems that need to perform an ACTIVATE of the configuration. If you have more than 16 systems in an SMS complex, you need to define the ACDS and COMMDS on a volume attached through a 3990 Model 6 storage control (the 3990 Model 3 does not have enough paths to make it possible to share attached volumes with more than 16 systems).

Prior to DFSMS/MVS 1.2, logical connectivity for all system-managed volumes and storage groups was controlled at the individual system level. Allocations, deletions, and accesses could only be performed on systems that had the logical (SMS and MVS) and physical (hardware) connectivity. This also applied to DFSMSHsm operations. Job failures would occur otherwise. In addition, the required catalogs needed to be accessible.

With DFSMS/MVS 1.2, when you define a volume or storage group to have connectivity to a system group, the volume or storage groups must be accessible to all systems that are part of the system group. Otherwise, job failures will occur.

When a common set of classes, groups, ACS routines, and a base configuration are applied across an MVS/ESA multisystem environment, the environment is a simple one. However, if SMS is not active on one of the systems, that system is not able to:

- Create data sets on system-managed volumes
- Delete system-managed data sets
- Extend system-managed data sets to new volumes
- Use JCL keywords supported by SMS

The COMMDS does not record DASD space usage changes for the system that has not activated SMS.

Using System Group Name with Pre-DFSMS/MVS 1.2.0 Systems

The system group name support may coexist with previous MVS/DFP and DFSMS/MVS releases which support SMS in the following situations:

- System group names are not exploited. If only system names are defined in the configuration, DFSMS/MVS 1.2.0 is SMS-compatible with previous releases of MVS/DFP or DFSMS/MVS.
- All pre-DFSMS/MVS 1.2.0 systems are explicitly defined in the configuration and only DFSMS/MVS 1.2.0 systems are represented by system group names. The pre-DFSMS/MVS 1.2.0 systems will interpret their name as a system name and will treat the system group names(s) as a system name. Only the DFSMS/MVS 1.2.0 systems (which contain the system group name support) will exploit the system group names. Refer to Figure 11 for an example of this coexistence environment.

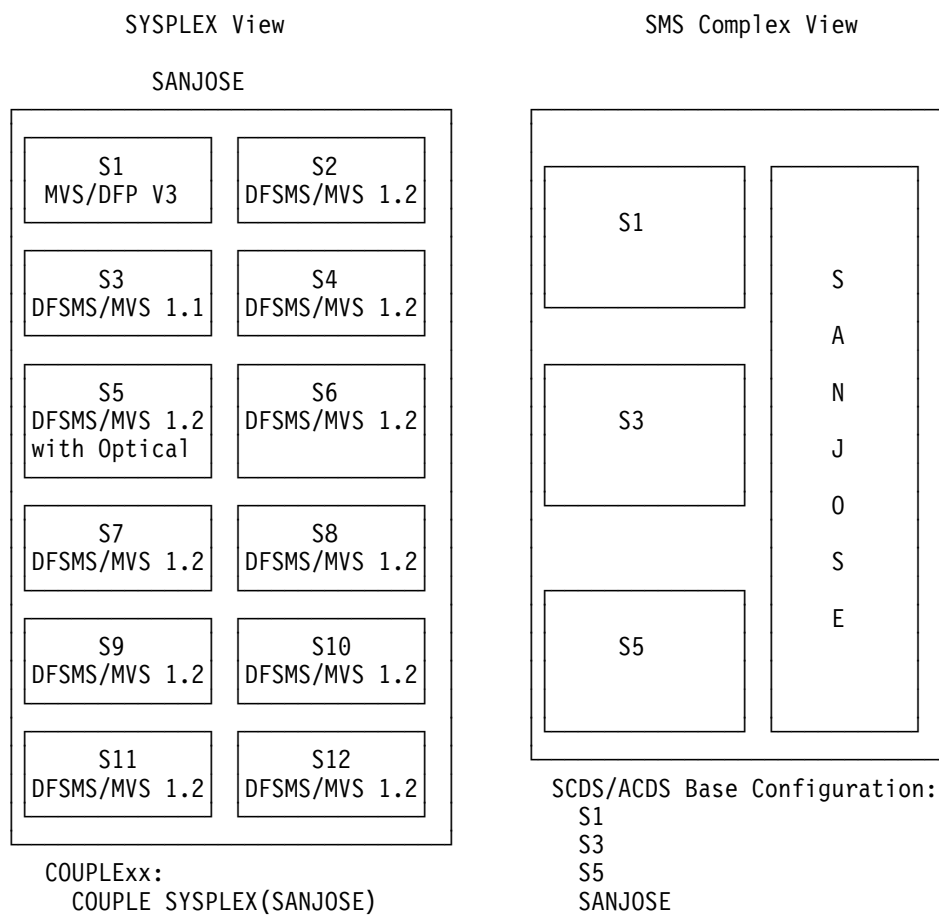


Figure 11. Coexistence when System Group Name Is Exploited

Note: The DFSMS/MVS 1.2.0 systems (which contain the system group name support) can coexist within the same SMS complex with either down-level systems or with systems using optical libraries for image processing applications.

Restrictions:

- You cannot use system group names in a JES3 environment.

- When defining an optical library in an SMS configuration, you can only connect it to one individual system and not to a system group.
- When defining a tape library, you can connect it to individual systems or system groups. However, to vary a tape library online or offline to a particular system, you must connect the tape library to individual system entries.

Using Different Releases of DFDSS, DFHSM, and DFSMS/MVS Programs

When converting to a new release of DFDSS or DFHSM, or to DFSMS/MVS, keep the following in mind:

- Although the DFSMSdss functional component of DFSMS/MVS can restore data dumped by DFDSS, you might not be able to use DFDSS to restore data dumped by DFSMSdss, particularly PDSEs dumped concurrently, extended sequential data sets, and VSAM data sets dumped using VALIDATE (default keyword). However, DFDSS can restore data dumped by DFSMSdss using VALIDATE if you have the VALIDATE PTF (support) on your DFDSS system.
- If you are going to have more than one release level of DFHSM running on shared systems, or you are going to have DFHSM and DFSMS/MVS with the DFSMSHsm functional component running on shared systems, apply the appropriate toleration PTFs to all versions of DFHSM to ensure data integrity and compatibility between the release levels and DFSMS/MVS.
- You cannot use the DFSMSHsm ARECOVER command with the INSTRUCTION or ACTIVITY parameters to recover the instruction data set or activity log of aggregate backups made with DFHSM. Otherwise, DFHSM aggregate backups can be recovered by DFSMSHsm. However, SMS attributes and storage requirements are not written to the ARECOVER activity log.

For more information on these compatibility considerations, see *DFSMS/MVS V1R2 Planning for Installation* and *DFSMS/MVS V1R2 DFSMSHsm Implementation and Customization Guide*.

Hardware Considerations

We recommend that you assign ranges of device numbers to specific device types. In a multi-system environment, you might have some devices defined on more than one system. This enables you to switch devices, such as user terminals, from the primary system to a backup system and the reverse.

You can have over 4096 unit control blocks (UCBs). If you have very large processors and large numbers of processors in your sysplex, you might need addressability to many devices.

The 3390 Model 9 offers more capacity at a lower cost than other 3390 models. While performance without cache might be acceptable for much of the data that is migrated to the 3390 Model 9, configuring with a cache-capable 3990 storage control can improve throughput and performance and also provide access to availability functions, such as concurrent copy and fast dual copy.

You can improve the throughput of your DASD subsystems by using the 3990 Model 6 Storage Control, with its increased cache sizes of up to one gigabyte. For some SMS complex configurations, you might need DASD with enhanced connectivity (an IBM 9340 or DASD attached to a 3990 Model 6 Storage Control).

| The 3995 Model 153 or 113 Optical Library Dataserver enables you to keep certain
| data online much longer than was previously affordable. You might also save
| money by putting OAM object backup copies on tape instead of optical.

| To take advantage of hardware compression for SAM and extended format VSAM
| data sets, the system uses host-based compression, such as on the IBM 9021
| 711-based models and 9121 511-based models, if available; otherwise, the system
| uses the slower Software Compression Facility.

Chapter 2. Planning DASD Storage Groups

Once you have evaluated the type of data used, and the types and amount of hardware you have available, you need to plan for the types of storage groups you will define.

In this chapter, we discuss the storage groups used with DASD volumes and data sets. We also discuss the storage classes you can define, and how the storage classes you define can affect the requirements for your storage groups and your hardware configuration. For example, many storage class attributes require that you use a cache-capable 3990 storage control.

Relating Your Storage Structure to Storage Groups

Before you implement system-managed storage, evaluate the tools and techniques you currently use for storage management. Then relate them to storage groups. If your DASD volumes are already pooled, it is easy to relate the volume pools to equivalent storage groups.

Unlike volume pools, storage groups do not require you to manage all the data within the storage group with a single management policy, nor are you required to use the same management policy for all the data sets on a single volume. With SMS, you can mix data sets with different management requirements on the same volume, providing for varying space, availability, and performance requirements within one storage group. You control these policies by assigning the data set the appropriate management and storage classes.

However, a system-managed data set can only reside in one storage group, even if it spans volumes. Ensure there are enough volumes in the storage group you assign to multivolume or extended sequential data sets when they are allocated, so that the allocation or subsequent extension of the data set does not fail.

Making a Smooth Transition to Storage Groups

Plan to implement DFSMS and the storage structure without disrupting data processing facilities vital to your business. *MVS/ESA SML: Implementing System-Managed Storage* discusses the conversion to the DFSMS environment and the coexistence of system-managed and non-system-managed data.

Maintain close contact with your user groups to make the conversion to the DFSMS environment as smooth as possible. Meet regularly with user group representatives while developing a schedule for implementation, to discuss your plans and any related information the users need to know.

Understanding DASD Storage Group Types

You can define three different types of storage groups for DASD volumes: dummy, pool, and VIO.

Dummy

A group of volumes that does not exist on the system, but which users can reference when they allocate existing data sets. This allows JCL that refers to a volume to continue to work after you remove the volume, move data off the volume, or change the volume's serial number. Thus, your users are less affected by changes to the hardware configuration, to the location of their data, or to volume serial numbers.

When allocating a data set, if a user indicates a volume that belongs to a dummy storage group, SMS locates the correct volume by using the catalog entry for the data set. However, a user cannot allocate a volume that belongs to a dummy storage group. Dummy volumes can only be used with data set allocations.

Do not add an existing volume to a dummy storage group, even if the volume is non-system-managed. Otherwise, you can cause data set allocation failures.

Pool

A group of existing DASD or optical (or both) volumes used for the allocation of new and existing system-managed data sets. Most storage groups are pool storage groups.

When you define a pool storage group, you define:

- Storage group status to your systems.
- System connectivity.
- Volumes that belong to the storage group.

When you choose the volumes for a pool storage group, you should consider how to provide the performance and availability requirements of the data assigned to the storage group. Remember to also consider device geometries and occupancy thresholds, system connectivity, and DFSMSHsm processing in a multihost environment.

- Requirements for automatic migration, backup, and dump processing.

These storage group attributes replace the ADDVOL statements used to identify non-system-managed volumes to DFSMSHsm for DFSMSHsm management. With storage groups, you do not need to use ADDVOL statements.

- Names of the systems that perform automatic space management, backup, and volume dump processing.
- The dump classes that DFSMSHsm should use for one or more full volume dumps.
- Thresholds used by allocation and DFSMSHsm to manage space on the volumes in the storage group.

VIO

A group used for new allocations that are to be directed to paging storage. The paging volumes are not actually defined to the storage group. VIO allocations are directed to the paging volumes defined to the system. These volumes are not system-managed, although the VIO allocations are system-managed. Use the VIOMAXSIZE attribute in the storage group definition to set a maximum size for data sets that are eligible for a VIO storage group.

Understanding DASD Storage Group Categories

This publication makes recommendations on how you should use *categories* of storage groups. A storage group category is a grouping of specific storage groups that contain the same type of data. For example, all small- and medium-sized primary data would be contained in the primary storage group category. This concept is analogous to storage pools in a non-system-managed environment.

The number of storage groups within a storage group category varies according to the types of devices you have. You should not mix devices with different geometries within the same storage group, because multivolume data sets cannot span volumes with different device geometries. Therefore, if you want to use both 3380 Model K and 3390 Model 2 devices in your primary storage group, you would need two specific storage groups in your primary storage group category.

You can, however, combine all 3390 look-alike devices (DASD and optical) in the same storage group. SMS is able to make the distinction during allocation so that you can mix devices of different performance characteristics within one storage group.

We use the following DASD pool storage group categories in the examples in this publication:

Primary Storage Groups

Contain most data sets.

Large Storage Groups

Contain especially large data sets, which would otherwise be placed in the primary storage group.

Database Storage Groups

Contain data used by a database management system, subdivided into categories for CICS, DB2, and IMS.

In addition, the VIO storage group is used for temporary data sets directed to paging storage rather than permanent DASD storage.

The Primary Storage Group Category

The primary storage groups should contain most of the data. Primary data includes the following types of data sets:

Interactive Data

Includes TSO user data sets, ISPF/PDF libraries, and any other data allocated and used during a terminal session.

Batch Data

Includes data sets generated in background jobs initiated either through an online facility such as TSO, or through specialized applications. Databases created through batch should be placed in a database storage group.

VSAM Data Sets

Includes all VSAM data sets that are not part of a database. VSAM data sets belonging to databases should be placed in a database storage group.

Multivolume Data Sets

Includes any data set that resides on more than one DASD volume. If you define large storage groups, some multivolume data sets might reside in the large storage group. The primary storage group should only contain small- to medium-sized multivolume data sets. See *MVS/ESA SML: Managing Data* for more information on allocating and managing multivolume data sets in a DFSMS environment.

Temporary Data Sets

Includes data sets that are created only for the duration of a job or job step. You can assign some temporary data sets to the VIO storage group. However, you cannot use VIO with VSAM data sets, so assign temporary VSAM data sets to the primary or large storage groups. You should also assign DFSORT temporary and work data sets to the primary or large storage groups, because VIO adversely affects DFSORT performance.

System Data

Includes the data sets used by MVS to keep the operating system running smoothly. Some system data sets can be system-managed; others should be placed on non-system-managed volumes.

System-Managed System Data Sets

System data sets can be placed on system-managed volumes if they are uniquely named. These data sets include:

- Master and user catalogs
- Page and swap data sets
- RACF data sets

Non-System-Managed System Data Sets

System programmers often maintain copies of active system data sets to install new products or product releases using the System Modification Program/Extended (SMP/E) maintenance program. Because such data sets are uncataloged, keep them on non-system-managed volumes. Data sets maintained by SMP/E include:

- SYS1.CMDLIB
- SYS1.HELP
- SYS1.LINKLIB
- SYS1.LPALIB
- SYS1.MACLIB
- SYS1.NUCLEUS

SYS1.PARMLIB
SYS1.PROCLIB
SYS1.SAMPLIB
SYS1.SVCLIB

The Large Storage Group Category

Large data sets that reside on DASD have special management requirements. Because these requirements can be provided at the data set level using a management class, you might place them in the primary storage group. However, you might want to place large data sets in a separate storage group because of:

- Possible space constraints for allocating new large data sets. The space occupied by large data sets can limit the free space available to other data sets.
- The need to maintain free space (a lower occupancy threshold) for large data sets to ensure that DFSMSHsm can recall them.

Whether you define a separate storage group for large data sets depends on the likelihood of these data sets causing allocation failures.

Figure 12 provides a list of recommended sizes to determine what constitutes a large data set, according to the type of DASD volume.

DASD Model	Minimum Data Set Size (in MBs)	Minimum Data Set Size (in Cylinders)
3380 Standard Models	65	90
3380 Model E	130	180
3380 Model K	190	270
3390 Model 1	95	115
3390 Model 2	190	225
3390 Model 3	285	335
3390 Model 9	855	1005
Note: Because of the unique performance characteristics of the 3390 Model 9, you might want to create a new storage group to ensure that applications needing traditional high-performance DASD are not allocated to the 3390 Model 9.		
9345 Model 1	100	145
9345 Model 2	150	220
Note: The MB and cylinder values are rounded up to the nearest multiple of 5.		

Figure 12. Minimum Size of Large Data Sets (by DASD Type)

Your definition of *large* should reflect the probability of a successful recall in five or fewer extents, based on the occupancy threshold you have set for the storage group. You define occupancy thresholds to control the amount of free space to be provided by DFSMSHsm space management processing. Ultimately, any data set that causes allocation problems during a recall should be considered large.

Even without a separate storage group for large data sets, you can use a management class to provide special treatment for them. For example, you can migrate large data sets directly to migration level 2 tape, rather than allow them to migrate to migration level 1 DASD. You can also use management class to prevent large data sets from migrating, which is a good solution if you have only a few large data sets that need to be immediately available on DASD. You should consider archiving seldom-used large data sets on tape.

An alternative to archiving to tape or DASD is to use the IBM 3995 Model 151 Optical Library Dataserver. You can keep these data sets online at just one-fifth the cost of DASD, and the retrieval rate is considerably faster than from tape storage. For more information on the 3995, see your IBM customer service representative.

If a large data set is causing performance problems, consider splitting the data set across multiple volumes, or converting it to a striped sequential data set. When deciding whether to split a data set across volumes, consider the increased complexity of backing up and recovering multivolume data sets against the potential of improving performance.

The Database Storage Group Categories

You might want to place databases in separate storage groups so that you can isolate the database applications from other users. This can help you maintain data integrity and ensure consistent performance for database applications.

Because you might want to place database data in separate storage groups, identify the amount of data currently managed by:

- Computer-Aided Design and Manufacturing System (CADAM**)
- Customer Information Control System (CICS/ESA)
- DATABASE 2 (DB2)
- Data Language/1 (DL/1)
- Information Management System (IMS/ESA*)
- INFO Management
- Additional application databases.

Consider creating a storage group for each database product you have installed. “Relating Volume Pools to Storage Groups” on page 29 discusses providing separate storage groups for CICS, DB2, and IMS data. *MVS/ESA SML: Managing Data* discusses using separate storage groups for database production and test data. Work closely with the database administrator when you plan the storage groups and classes for this data.

If you use generation data sets for batch databases, you can include them in the primary storage group. You can choose special management criteria for such data sets by assigning them special management classes. For more information on managing generation data sets, see *MVS/ESA SML: Managing Data*.

The VIO Storage Group

You can allocate most temporary data sets as VIO data sets to improve performance. You can define a VIO storage group and assign temporary data sets to it. However, direct temporary VSAM data sets to a pool storage group rather than to a VIO storage group because VSAM data sets cannot use VIO. Because the system prevents VSAM data sets from being allocated to VIO, you must provide a pool storage group to which the system can direct the VSAM allocation in your storage group ACS routine.

Although you can direct DFSORT data sets to VIO, we do not recommend that you allocate DFSORT temporary output and work data sets as VIO data sets. VIO reduces elapsed time at the cost of increased CPU time for DFSORT applications.

To ensure that all temporary data sets are system-managed effectively, design your storage class and storage group ACS routines so that the storage class ACS routine assigns all DFSORT temporary data sets to a non-VIO storage class. See Figure 25 on page 60 for an example of an ACS routine that does this.

You can then allocate the remaining temporary data sets to the primary storage group or to a VIO storage group. You choose the size limit of data sets that are allocated to the VIO storage group in the storage group definition. This allows you to make more effective use of available space and provide better performance for temporary data sets.

Relating Non-Pooled Volumes to Storage Groups

If you do not use volume pools, that is, your DASD volumes are privately controlled by your user groups, you do not need to create them before implementing system-managed storage. Follow the procedure outlined in *MVS/ESA SML: Implementing System-Managed Storage* to convert non-pooled volumes to system-managed volumes.

Figure 13 on page 28 shows an example of mapping the data sets on private volumes to storage groups.

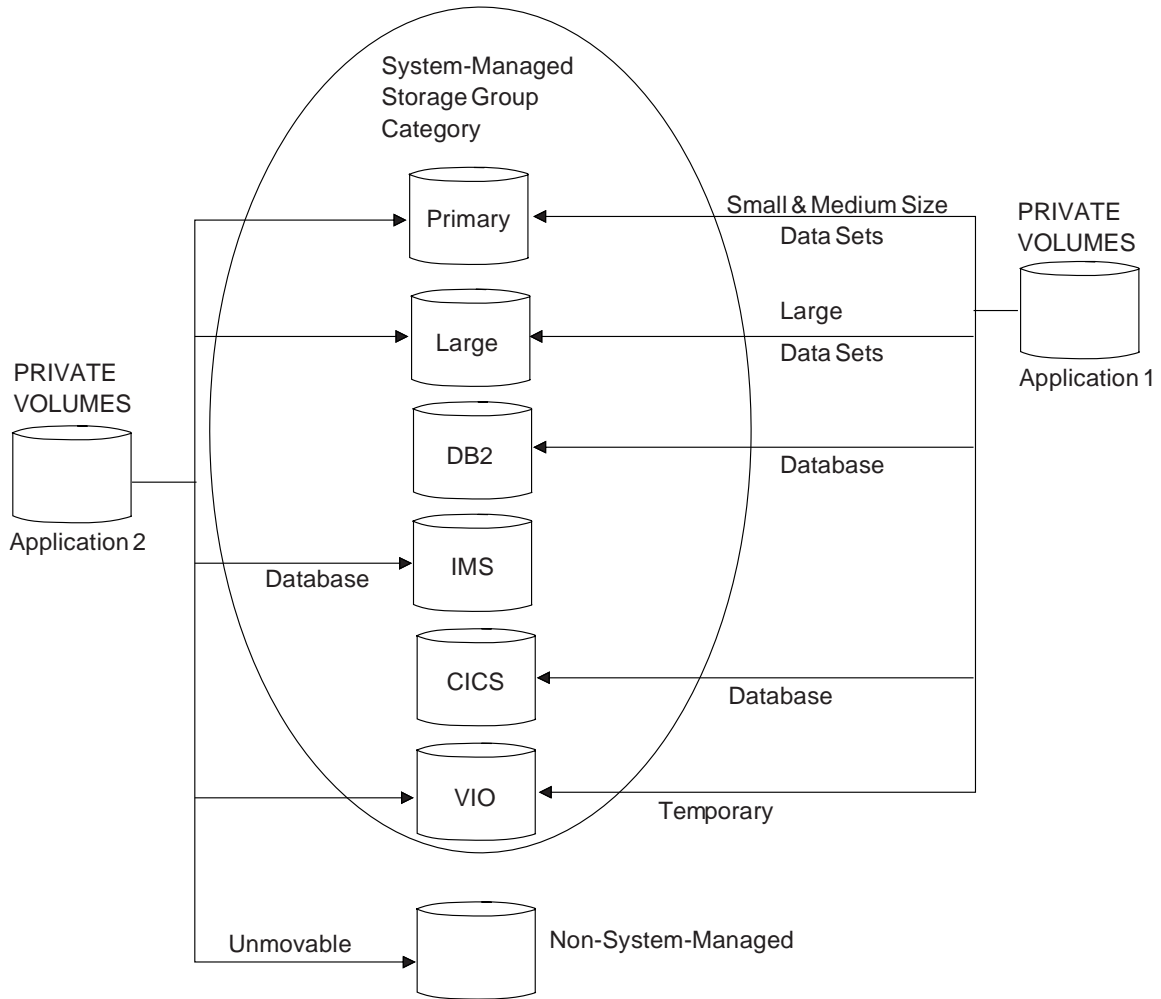


Figure 13. Sample Mapping of Private Volumes to Storage Groups. The data sets on the private volumes are split among the appropriate storage groups according to the type of data set.

One of the tasks you must complete to make your conversion is to analyze the data on private volumes and map the data to appropriate storage groups. You can use the ISMF Data Set application to create lists of data sets, based on various criteria, for example, data set organization. With these lists, you can identify data that cannot be system-managed, and plan how to support them. You can also determine what types of data you have that can be system-managed, and use that information in planning your storage groups.

Plan to convert DASD data sets that cannot be system-managed. In the meantime, group unmovable data sets and other unsupported data types on volumes that are not system-managed. Establish a system standard so that all unmovable data sets are clearly marked, either through data set name or organization, as unmovable. This helps to eliminate the danger of accidentally moving any data that has specific device dependencies.

For information on identifying types of data that cannot be system-managed, and on methods for converting to supported types of data, see *MVS/ESA SML: Managing Data*.

Relating Volume Pools to Storage Groups

If you have already set up a storage pooling structure, you can convert your existing volume pools to equivalent storage groups. However, some volume pools needed in a non-system-managed environment are not necessary in a system-managed environment. For example, there is no reason to separate VSAM data sets from other data sets in storage groups, although you can have a separate volume pool specifically for VSAM data sets.

Figure 14 on page 30 shows an example of how to map volume pools to storage groups and unmanaged volumes. Figure 15 on page 30 explains how the data sets in the volume pools are divided among the storage groups.

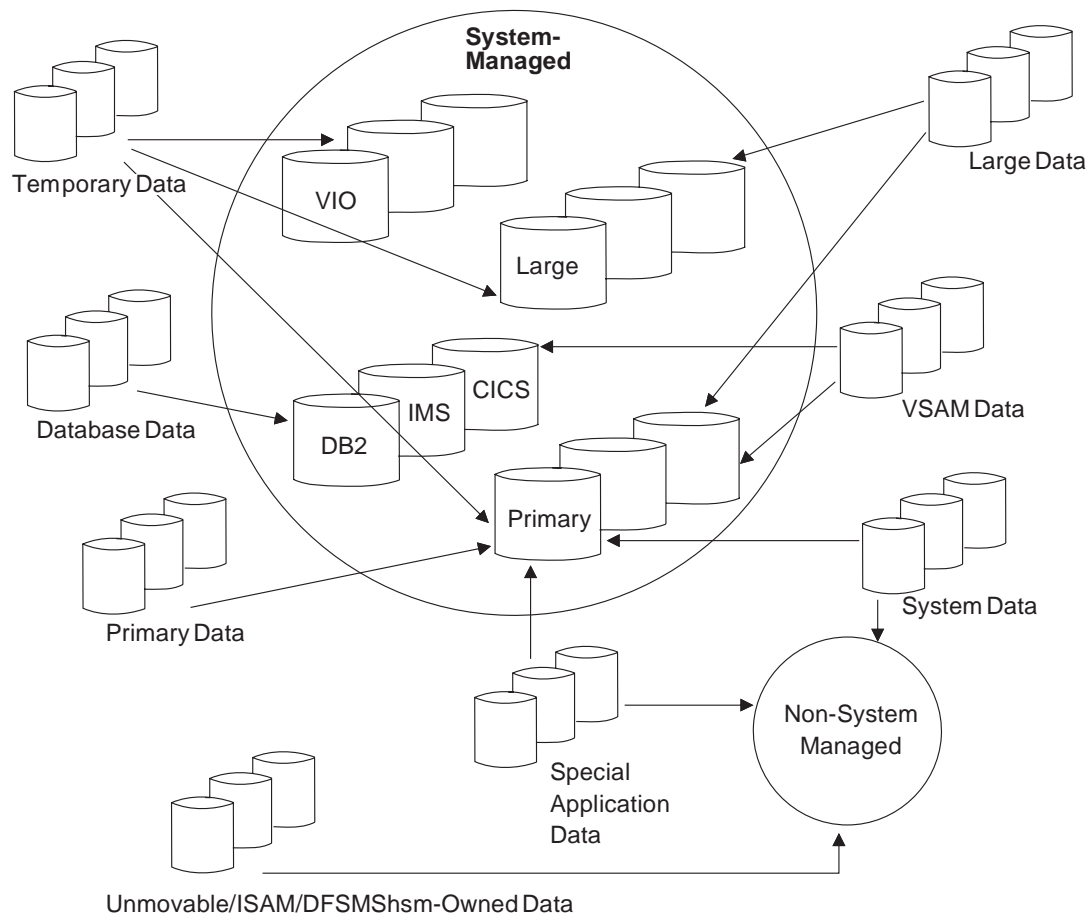


Figure 14. Sample Mapping of Volume Pools to Storage Groups. Most data is placed in the primary storage group, except for very large data sets, which are placed in the large storage group. Database data is split between the CICS, DB2, and IMS storage groups. Selected temporary data sets are placed in the VIO storage group. See Figure 15 on page 30 for an explanation of the mapping between the volume pools and storage groups.

Figure 15. Relationship of Volume Pools to Categories of Storage Groups

Volume Pool (non-SMS)	Description	Storage Group Category
Database	CICS, IMS, and DB2 databases	Database
Large	Large interactive, batch, or temporary data sets	Large or primary
Primary	Interactive, batch, and temporary data sets	Primary
Special Application	For isolating key applications from other applications, device failure, or for security reasons	System-managed in primary
System	System and control data sets	System-managed in primary
Temporary	Temporary non-VSAM data sets	VIO, primary, or large
VSAM	VSAM data sets, some possibly belonging to databases	Primary or database

You can use the ACS routines to filter on the size of the data set to direct it to the appropriate storage group. The storage group ACS routine shown in Figure 24 on page 58 shows an example of filtering on the size of a data set.

Determining DASD Storage Group Requirements

Setting up your system-managed storage structure is an iterative process. Once you have determined the storage groups you need, consider the performance and space requirements of the system-managed data in each storage group. As you plan for data, management, and storage classes, you might find common needs among your storage groups. Because SMS allows you to manage individual data sets rather than volumes, you use the same storage classes for all storage groups. Keep your entire storage structure in mind, and modify your planned system-managed storage structure until you have a set of storage group, storage class, and management class requirements that characterize your data needs.

Determining the Number of Storage Groups You Need

Strive to keep your storage groups and storage group categories to a minimum. Because you can use storage and management class to account for different backup, performance, and availability requirements for data sets in the same storage group, you can combine many data sets you now have in different volume pools into one storage group, without creating different storage group categories.

The fewer storage groups you define, the easier it is for SMS to effectively balance performance and space usage across the system. If you have too many storage groups, or too few volumes in any one storage group, you are less able to fully use the space and performance advantages of SMS. A large number of storage groups can make the SMS configuration too complex and difficult to maintain.

Because the access methods cannot handle multivolume data sets defined across volumes with differing device geometries, do not mix volumes with different device geometries in the same storage group. However, you can mix device geometries within the same storage group category.

For example, you could define two storage groups and treat them as being one primary storage group: PRIME80, containing 3380 volumes; and PRIME90, containing 3390 volumes. To do this, always assign both storage groups together in your storage group ACS routines.

The best approach is to set up a primary storage group category containing as much of your data and as many of your devices as possible, and secondary storage groups to handle data with specialized requirements or management techniques. Within the primary storage group category, define a storage group for each device geometry.

Using Special Purpose Storage Groups

To determine how many storage group categories you need, consider the types of data used and your existing volume pools. Know how the data is used, and whether there are requirements to isolate some data from other data.

Use ISMF to list all data sets and classify them according to data type. Work with the organizations you support to determine special storage needs, and identify data that requires separate treatment.

Once this is done, you can estimate the amount of space required for each storage group. For more information on creating data set lists with ISMF, see *MVS/ESA SML: Managing Data* and *DFSMS/MVS V1R2 Using ISMF*.

The following list contains some reasons you might consider for creating special storage groups in addition to the primary storage group.

Key application isolation

If you have an application that you want to isolate from other applications, you can define a special storage group for it. You can also use storage groups to limit the impact of a device failure to a specific application or to isolate security-sensitive data.

Special service requirements

If you use database data, you can set up separate storage groups to maintain database integrity and to provide for special backup and recovery requirements. See *MVS/ESA SML: Managing Data* for more information about managing database data.

If you have some very large data sets, you can define a separate storage group for them. This can help you avoid space allocation problems in the primary storage group.

Consistent processor connectivity

You can use storage groups to define the connectivity of volumes to specific hosts, where you have multihost access to application volumes.

Other business reasons

You can define a separate storage group for an application in unique situations. For example, you can create a separate storage group for batch applications that require large amounts of space to process.

Device Geometry and Capacity in Storage Groups

Although not required, we recommend that you define pool storage groups so that they only contain devices of the same geometry. The device *geometry* is the track size and number of tracks per cylinder for the device.

By defining pool storage groups so that the device geometry is the same for all volumes in the storage group, you can ensure that volumes of the same geometry are available when multivolume data sets need to extend to new volumes. Access method services cannot handle multivolume data sets defined across volumes with differing device geometries.

For example, if you want to use 3380 and 3390 devices in your primary storage group category, you should define two primary storage groups: one containing 3380 devices, and another containing 3390 devices.

Because 3390 devices in 3380 track compatibility mode are geometrically the same as 3380 devices, you can combine these devices in a single storage group. Because the 3390 devices are in 3380 track compatibility mode, the access methods see them as 3380 devices.

Although you should separate devices according to geometry, you do not need to separate them according to capacity. For example, you can combine all models of the 3390 into a single storage group. The different capacities only affect volume thresholds, as is explained in “Setting Thresholds for Storage Groups” on page 50.

See Figure 24 on page 58 for an example storage group ACS routine that assigns storage groups containing different device geometries. You can use the ACS routine to select one or all of these storage groups for allocating data sets that should reside on primary storage volumes.

Combining Device Types in Storage Groups

SMS can closely match a performance request given in the storage class. Therefore, you can combine devices of different performance characteristics within one storage group. However, the types of devices must appear the same. For example, you can mix all 3390 look-alike devices, DASD and optical, in the same storage group.

Combining mixed devices in one storage group helps minimize the number of storage groups required. It also allows you to fully exploit the performance advantages of SMS.

DASD Volume Naming Conventions

Once you have decided which storage groups you need, you can develop a naming convention that identifies the correlation between a storage group and volume serial numbers of the volumes belonging to the storage group. You should also develop a naming convention for your non-system-managed volumes, to help identify their function.

Naming conventions allow you to use the filtering techniques provided by ISMF and DFSMSdss more easily.

Figure 16 on page 34 shows some examples of naming conventions for both system-managed and non-system-managed volumes. The volume serial numbers use prefixes similar to the storage group names. For example, all volume serial numbers beginning with the letters **PRM** belong to the primary storage group category. You can name the volumes PRM001, PRM002, PRM003, and so forth.

Figure 16. Volume Naming Conventions

Volume Serial Number Prefix	Description	Storage Group Category
System-Managed Volumes		
CIC	CICS volumes in the database storage group	Database
DB2	DB2 volumes in the database storage group	Database
HBU	DFSMSHsm DASD backup volumes	None
IMS	IMS volumes in the database storage group	Database
LRG	Volumes in the large storage group	Large
ML1	DFSMSHsm migration level 1 volumes	None
PRM	Storage volumes in the primary storage group	Primary
SYS	Other system volumes	Primary, large, or non-system-managed volumes
Non-System-Managed Volumes		
PAG	System paging volumes	None
SPL	System spool volumes	None
UNM	Volumes containing unmovable data sets.	None

You assign volume serial numbers (volser) to DASD volumes when you initialize them. Make sure that you do not create a volume with the same volser as an existing DASD or tape volume. For information on volume initialization, see “Preparing the Volumes for Storage Groups” on page 43.

Storage Group Naming Conventions

As discussed in “Understanding DASD Storage Group Categories” on page 23, you can classify your storage groups according to storage group category. To simplify the task of managing storage groups, and writing and understanding your ACS routines, you need to develop naming conventions for your storage groups. With naming conventions, you can quickly identify the storage group category that contains a specific storage group.

If you have only one type of DASD, you probably only require one storage group within each storage group category. However, as you add DASD types, you might need to add storage groups. For recommendations on how to best handle different device types and geometries within storage groups, see “Device Geometry and Capacity in Storage Groups” on page 32.

We recommend you use the following naming conventions for storage groups:

Storage Group Category	Storage Group Name
Primary	PRIMExxx
Large	LARGExxx
CICS database	CICSxxxx
DB2 database	DB2xxxxx
IMS database	IMSxxxxx

Storage Group Category	Storage Group Name
VIO	VIO

Note: Replace the x string with something you find meaningful. For example, PRIME90 could be the name of a primary storage group containing 3390 DASD volumes, PRIME80 one containing 3380 DASD volumes.

Figure 17 shows an example of using both storage group and volume serial number naming conventions within the primary storage group category.

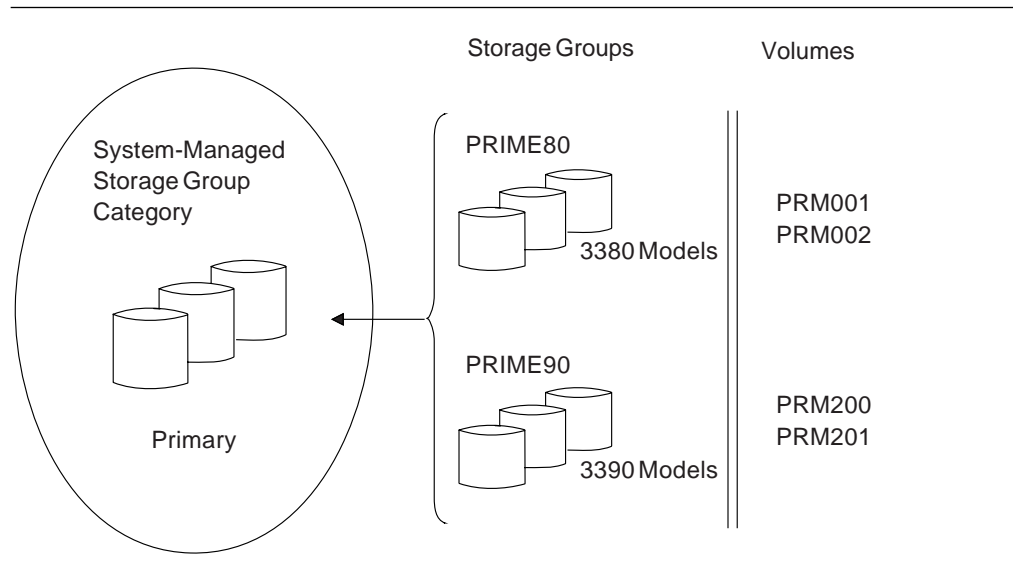


Figure 17. Storage Groups and Volumes within the Primary Storage Group. The primary storage group category contains two storage groups. PRIME80 contains 3380 DASD volumes (all models) with volume serial numbers in the range PRM001 to PRM199. PRIME90 contains 3390 DASD volumes (all models) with volume serial numbers in the range PRM200 to PRM299.

Sharing DASD Volumes and Storage Groups

If you have more than one system, you need to consider the sharing requirements for your data. As long as you use global resource serialization (GRS), which converts hardware reserves and releases into data set enqueues and dequeues, you should not have problems with poor performance due to users being unable to access data on a shared volume. For more information, see *Planning: Global Resource Serialization*.

Sharing DASD volumes and storage groups can improve availability and load balancing. For example, DFSMSHsm can use the least active system to do its space management. However, you might not want to share all data. For example, you might decide to shield a database from other systems.

For simplicity, if you share any volumes within a storage group between systems, share all the volumes in the storage group. Base your data sharing on a system or system complex, not on individual volumes.

One effective approach is to share all primary and large storage groups between all systems. Then, create special storage groups for applications that you want to only be available on one or a subset of systems.

By basing data sharing on storage groups instead of volumes, it is easier to maintain your system and identify how a volume is being used.

However, you should never share system-managed volumes with a system using MVS/XA DFP Version 2. It cannot recognize SMS information, nor can it update SMS information. An MVS/XA DFP Version 2 system corrupts catalog entries for system-managed data sets if it tries to update the entry. If you must share volumes with an MVS/XA DFP Version 2 system, only share non-system-managed volumes.

If you share catalogs or control data sets, be sure to account for heavy I/O rates in your performance projections. Use storage classes to place shared data that is frequently accessed by multiple systems on the highest performance devices.

See *DFSMS/MVS V1R2 DFSMSHsm Storage Administration Guide* for configuration information related to sharing volumes managed by DFSMSHsm for data set backup and migration.

An Availability Tip for Volumes in a Multihost Environment

To improve availability for volumes normally accessed from only one system, you can identify a device as shared during system initialization and then disable it to all but one of the systems defined to the volume. This allows you to easily switch devices from one system to another in case of a control unit or system failure.

If you are sharing resources across MVS systems in a system complex, we recommend that you allow the system-managed volumes and storage groups to be accessible from all systems that are part of the system group. This minimizes the number of job failures that can occur.

Planning for the Number of Volumes

When you plan your storage groups, determine how many DASD volumes you need for each storage group by considering the amount of space required and the capacity of the available devices. You can add new volumes directly to storage groups, or add existing volumes to storage groups and convert them.

“Converting Existing Volumes” on page 46 explains how to convert volumes. Plan to reclaim any private volumes from user groups as data is moved from the volumes into storage groups.

When planning for the number of volumes you need, include the DASD volumes needed for DFSMSHsm migration level 1 data. By expanding the scope of the storage that DFSMSHsm manages automatically, you might find that additional migration volumes are required.

You should also provide spare volumes for recovery of system-managed volumes. In addition, DASD space is required for OAM table spaces for object directories and storage.

Configuring Pool Storage Groups for Performance and Availability

When you define your storage groups, you need to consider your requirements for performance and availability. You establish performance and availability goals by creating appropriate storage classes.

However, even if you define storage classes with specific performance and availability goals, those goals can be met only if you configure your storage groups so that they contain the hardware devices required to meet those goals. For example, many performance and availability goals can only be met if you use cache-capable storage controls with the Extended Platform.

Using Storage Class to Define Performance and Availability Goals

With SMS, you use storage classes to set performance and availability goals for your data. By defining a variety of storage classes, you can provide for varying levels of performance and availability according to the needs of your business and the service level agreements with your user groups.

Storage classes enable you to:

- Separate the logical requirements for performance and availability for data sets from their physical placement.
- Separate the placement of objects from the physical characteristics of devices.
- Reduce the need to manually place data to meet these particular requirements.
- Effectively use the performance capability of storage devices.
- Define availability and accessibility attributes for data sets.

You use ISMF to define storage classes. Each system-managed data set is assigned a storage class when the data set is created. Although not recommended, users can choose the storage class for data sets with JCL. You can override any user-chosen storage class in your storage class ACS routine.

When a system-managed data set is created, the ACS routines assign the data set to a storage group. When selecting the candidate volume from the storage groups, SMS uses the values of the attributes given in the storage class that is assigned to the data set.

Evaluate which hardware devices you need in a storage group to meet the performance objectives represented by the storage classes you define. When you are defining storage class values that require a high performance device, make sure that the hardware devices that are assigned these objectives are capable of delivering the performance.

For information on selecting the storage class attributes, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

Using Cache in the DFSMS Environment

SMS uses the response time given in a data set's MILLISECOND RESPONSE storage class attribute to select volumes that can meet the response time requirement, if the volumes are available. If they are not, SMS selects volumes that have a performance capability that most closely matches the given response requirement.

SMS can select a cache- or fastwrite-capable volume to meet the requested response.

SMS uses caching provided by cache-capable storage controls.

Using Cache-Capable 3990 Storage Controls

The cache-capable 3990 storage controls provide many functions beyond the basic read caching available in earlier storage controls. In addition, SMS increases the benefits available for system-managed data sets.

For example, SMS determines which data sets are cached independently of the volume on which the data set resides. This allows you to:

- Control which data sets are cached
- Provide fast I/O response time for selected data sets
- Mix cached and non-cached data sets on the same volume to fully use DASD capacity

By using SMS to manage data sets on DASD behind a 3990 storage control, you can:

- Take advantage of all the 3990 extended functions
- Reduce the need to monitor and tune the cache
- Ensure cache and DASD fast write support for performance critical data

Be aware that mixing system-managed and non-system-managed volumes behind a 3990 can affect SMS cache usage. SMS controls only the caching of data sets on system-managed volumes. Because all the data on a non-system-managed volume is eligible for caching if you have enabled caching for the volume, the amount of 3990 storage control services available to the system-managed data might be reduced.

The 3990 storage control retains, by volume, the status of its features for either an initial microcode load (IML) of the 3990 unit, or host system initial program load (IPL), whereas the 3880 Models 13 and 23 do not. This eliminates the need to re-establish the feature status following a service interruption.

See *3990 Planning, Installation, and Storage Administration Guide* for information on using the SETCACHE command to control the caching characteristics of the 3990 storage controls.

Using Dynamic Cache Management

SMS uses caching statistics provided by cache-capable 3990s to control the use of cache and nonvolatile storage for system-managed data sets. Depending on the load on the cache and nonvolatile storage, SMS allows or inhibits caching for individual data sets. Thus, at I/O initiation, SMS can suppress or allow caching for data sets that have storage classes without a cache requirement. This is called *dynamic cache management*.

When a millisecond response time (MSR) is not given in the storage class, SMS selects the volume that is closest to the performance of a 3390-3 with cache active. The closest available volume might not be cache-capable or have cache active.

To use dynamic cache management, do not define an MSR value in the storage class definition. Data sets without an MSR value are classified as *may-cache* data sets. The system determines whether to cache may-cache data sets when they are used. If most data sets are given a may-cache storage class, the system can more easily balance cache usage and performance.

To request that the system is to cache a data set, assign the data set a storage class with a very low MSR value. However, this does not always guarantee that the data set is cached. To prevent the system from using cache, use an MSR value of 999.

SMS automatically provides dynamic cache management for volumes attached through a cache-capable 3990 storage control. Dynamic cache management allows broader usage of the cache and DASD fast write features when they are not fully used. If they are being overused, it can restrict their usage.

With DFSMS/MVS, if you are using a cache-capable 3990 storage control with Extended Platform, the system evaluates how effectively a data set is using cache, and prevents it from using cache if it is not a good cache candidate. This allows as many data sets as possible to benefit from the cache. To take advantage of this capability, set the MSR values according to the priority of the use of the data set, rather than on whether the data set is a good cache candidate. The enhanced dynamic cache management used in the Extended Platform helps prevent cache over-use problems before they occur, rather than reacting to problems as they occur.

Evaluating Cache Usage

DFSMS/MVS collects cache usage statistics from data sets cached in cache-capable 3990 storage controls. The system writes these statistics to the SMF data set as type 42 records.

The type 42 subtype 5 record contains information on response time statistics according to storage class. The type 42 subtype 6 record contains information on data set caching statistics.

You can use these SMF records to help you evaluate cache usage and performance. For more information about these SMF records, see *MVS/ESA SPL: SMF*.

Using Striping to Improve Sequential Processing Performance

You might have some large physical sequential data sets with high I/O rates that could benefit from using striping. *Sequential data striping* is a technique where data for one data set is interleaved among multiple devices. Striping is used to transfer data between DASD and memory at a faster rate than an individual DASD can handle. In DFSMS/MVS, sequential data striping is restricted to extended sequential data sets.

All stripes of an extended sequential data set must reside on volumes attached through cache-capable 3990 storage controls with Extended Platform using ESCON* channels. To benefit from the increased I/O rate possible with striping, you need to ensure that the storage group assigned to the data set has enough volumes connected through different paths and storage controls so that each stripe can be accessed without interfering with access to the other stripes.

If you are using DFSMS/MVS, you can use extended sequential data sets with BSAM or QSAM. For most applications, there is no difference between processing a physical sequential data set and an extended sequential data set. Thus, you can convert data to extended sequential without making major changes to your applications, if any changes are required at all.

To define an extended sequential data set, the data set must be assigned a data class with the DSNTYPE attribute equal to ER or EC. ER designates *extended required*: if the data set cannot be striped, allocation fails. EC (*extended preferred*) specifies that if the data set cannot be striped, it is allocated as non-striped. With either designation, the system attempts to meet the requested number of stripes. If it cannot, a smaller number is used, and the allocation does not fail.

The storage class assigned to a striped data set is used to determine the number of stripes allocated to the data set. Each stripe is allocated on a separate volume, and the system tries to separate the stripes behind as many different storage control units as possible. An extended sequential data set can have as many as 16 stripes.

The system uses two storage class attributes to calculate the number of stripes: SUSTAINED DATA RATE and GUARANTEED SPACE. Use SUSTAINED DATA RATE to set a performance objective, in MBs per second, for a data set. The system uses your performance objective, and the transfer rate of the DASD, to determine a target number of stripes to allocate. The actual number of stripes allocated depends on the available volumes. Use GUARANTEED SPACE if you know the number of stripes that are required. If the storage class has YES for GUARANTEED SPACE, the system uses the volume count on the allocation request as the number of stripes. If volume serial numbers are included, they are used in the order given.

Configuring for Data Availability

You might have some data that must be continuously available to meet your business needs. For example, databases used to fill orders can be needed 24 hours a day.

You can use storage class attributes and related hardware to help meet the data availability needs of your business.

Using Dual Copy for Critical Data

If you have data that is especially critical, you can create a storage class with an AVAILABILITY value of CONTINUOUS.

If a data set is assigned a storage class requiring continuous availability, the data set is allocated on a dual copy DASD volume. A *dual copy* volume has another DASD volume defined as a duplicate copy. Any updates to the volume are also made on the second volume. The second volume is defined with the same volume serial number, but it is offline to the system. If a failure occurs on the primary volume, the system automatically switches to the secondary volume, and you can continue using the data set.

Use the access method services SETCACHE command to define a dual copy pair. The primary and secondary volumes of the dual copy do not need to be on the same string, they must only be accessible from the same cache-capable 3990 storage control. We recommend that you attach the secondary volume through a different storage director and channel than the primary volume. This isolates the dual copy pairs and minimizes the effect of a failure in the storage control.

You must ensure that dual copy primary volumes are defined to the storage group that is assigned to a data set with a storage class indicating continuous availability. If no dual copy primary volumes are available in the storage group, the data set allocation fails. Allocations can also fail if there is not enough space on dual copy volumes to satisfy the space requirements.

Only data sets assigned a storage class requesting continuous availability are allocated on dual copy volumes. Data sets assigned storage classes requesting standard availability are only allocated on volumes that are not defined as dual copy.

For more information on defining and using dual copy volumes, see *3990 Planning, Installation, and Storage Administration Guide*.

Using DFSMSdss to Concurrently Copy Critical Data

With the DFSMSdss functional component of DFSMS/MVS, you can have copies of data sets made concurrently. This allows the system to copy the data set while other users are updating the original copy. The system keeps track of changes made to the data set and ensures that the copy is equivalent to the original data set at the time the copy was initiated. You can use storage class and management class to control which data sets can be copied concurrently.

To concurrently copy a data set, the data set must reside on a volume connected through a cache-capable storage control with Extended Platform. All volumes behind a storage control with Extended Platform are considered concurrent copy volumes by SMS. The storage class ACCESSIBILITY attribute controls whether the data set is allocated to a volume capable of using concurrent copy.

If you set the ACCESSIBILITY attribute to REQUIRED, the data set can only be allocated on a volume capable of using concurrent copy. If no volumes are available, or if there is not enough available space on eligible volumes, the allocation fails.

If you set the ACCESSIBILITY attribute to PREFERRED, the data set is allocated on a volume capable of concurrent copy if one is available and it contains sufficient space for the allocation. If a concurrent copy volume is not available, the data set can be allocated on a non-concurrent copy volume.

If you set the ACCESSIBILITY attribute to STANDARD, the system first tries to allocate the data set on a volume not capable of using concurrent copy. However, if none are available, the data set might be allocated on a volume capable of concurrent copy.

If you set the ACCESSIBILITY attribute to NOPREF, the system allocates the data set on a volume regardless of whether the volume supports concurrent copy.

You can also use the management class BACKUP COPY TECHNIQUE attribute to have DFSMSHsm use concurrent copy during backup processing.

For more information on making concurrent copies, see *DFSMS/MVS V1R2 DFSMSdss Storage Administration Reference*.

Chapter 3. Implementing DASD Storage Groups

After you have planned the storage group categories and storage classes you require, you need to define your storage groups and write the storage group and storage class ACS routines.

This chapter focuses on implementing and working with storage groups. Although we discuss some points concerning storage class here, for more complete information on storage class see *MVS/ESA SML: Managing Data*.

Preparing the Volumes for Storage Groups

As you add DASD volumes, add them directly to storage groups unless you are going to use them as DFSMSHsm volumes. You should also convert your existing non-system-managed volumes to system-managed as they become available.

Defining the Hardware Configuration

You can dynamically add devices to the hardware configuration or delete them from it without a system interruption. Using the Hardware Configuration Definition*, you can install or remove control units, channel paths, DASD volumes, tape devices and unit record devices without an initial program load (IPL) of the operating system or a power-on-reset of the hardware. This removes a constraint on hardware installation and reduces the hardware planning effort.

For information on how to use the Hardware Configuration Definition to dynamically change the I/O configuration, see *HCD: Planning and Hardware Configuration Definition: User's Guide*.

Adding New Volumes

To add a new volume, use the following procedure:

1. Add the device to the hardware configuration using the Hardware Configuration Definition.
2. Define the new volume to a storage group using the ISMF Storage Group Application, and define the volume's status to your systems.
3. Activate the new SMS configuration.
4. Initialize the volume with the ICKDSF INIT command in offline mode with minimal initialization, specifying the STORAGEGROUP and INDEX parameters. You need to initialize it in offline mode because a volume does not have a label when it is first installed.

Figure 18 on page 44 shows an example of initializing a system-managed volume in the PRIME storage group using ISMF panels. For more information on initializing DASD volumes, see *ICKDSF User's Guide and Reference*.

5. Vary the volume online to MVS.

```

                                INIT ENTRY PANEL
COMMAND ==>

VOLUME SERIAL: PRM020           STORAGE GROUP: PRIME
MVS STATUS:    ON

FOR VOLUME TABLE OF CONTENTS, SPECIFY LOCATION AND SIZE:
CYLINDER ==> 3                   (0 to 9999)
TRACK     ==> 0                   (0 to 9999)
SIZE     ==> 45                   (0 to 99999 tracks)

WANT TO GENERATE INDEX           ==> (Y or N)
WANT VOLUME TO BE SMS MANAGED   ==> Y (Y or N)

FOR VTOC INDEX, SPECIFY LOCATION AND SIZE:
CYLINDER ==> 1                   (0 to 99999)
TRACK     ==> 1                   (0 to 99999)
SIZE     ==> 29                   (0 to 99999 tracks)

SPECIFY THE FOLLOWING TO BE WRITTEN ON THE VOLUME LABEL
VOLUME SERIAL   ==> PRM020       (1 to 6 chars)
OWNER ID        ==>              (1 to 14 chars)
USE ENTER TO PERFORM SELECTION;
USE HELP COMMAND FOR HELP; USE END OR CANCEL COMMAND TO EXIT.

```

```

                                INIT ENTRY PANEL
COMMAND ==>

VOLUME SERIAL: PRM020           STORAGE GROUP: PRIME
MVS STATUS:    ON

TYPE OF INITIALIZATION          ==> 1          (1 - MIN, 2 - MED, 3 - MAX)

FOR MEDIAL INITIALIZATION, SPECIFY:
WRITE DATA RECORDS            ==>           (Y or N)

PURGE UNEXPIRED DATA SETS     ==> N         (Y or N)

RESUME FROM THE LAST CHECKPOINTED LOCATION ==> Y (Y or N)
PRINT A VOLUME MAP DURING INITIALIZATION  ==> Y (Y or N)

IF MAXIMAL INITIALIZATION, SPECIFY:
NUMBER OF TIMES EACH TRACK TO BE CHECKED ==> (1 to 10)
RECLAIM TRACKS PREVIOUSLY DEFECTIVE      ==> (Y or N)
USE I/O DELAY                            ==> N (Y or N)
VERIFY VOLUME SERIAL                     ==> 1 (1 - YES, 2 - NO, 3 - NONE )

USE ENTER TO PERFORM SELECTION; USE END COMMAND TO VIEW PREVIOUS PANELS;
USE HELP COMMAND FOR HELP; USE CANCEL COMMAND TO EXIT.

```

Figure 18. Example of Initializing a New Volume. You can use the ISMF Volume or Storage Group Application to create ICKDSF jobs for volume initialization.

Determining VTOC Placement

Historically, the placement of the volume table of contents (VTOC) on a volume has been an important performance consideration, because seek times were considerably slower on older devices than they are on 3380, 3390, and 9345 DASD. However, with faster devices and cached storage controls, you do not need to consider VTOC placement for improving performance.

For volumes in the large storage group, you should place the indexed VTOC at the beginning of the volume to avoid unnecessary space fragmentation.

Wherever you decide to place the VTOC, ensure that the VTOC and the VTOC index occupy contiguous space to reduce search time.

Estimating VTOC and VTOC Index Size

The volume table of contents cannot be extended, so it is important that you make an accurate estimate of the size the VTOC should be for the volume.

Estimate the size of the VTOC by estimating the maximum number of data sets to be stored on the volume, and dividing that value by the number of data set control blocks (DSCBs) that can reside on a track of the device. When you estimate the number of data sets, count VSAM key-sequenced data sets as two data sets, because the index and data component each require a DSCB.

For example, a track on a 3390 device can hold 50 DSCBs. If you estimate that the volume will contain 1000 single-extent data sets, the VTOC size should be at least 20 tracks.

The size of the VTOC index depends on the number of data sets and the length of the data set names.

Use the formulas in *ICKDSF User's Guide and Reference* to calculate the size of the VTOC and the VTOC index. Be generous in your estimates to allow for extending data sets.

VTOCs and VTOC indexes for DFSMSHsm migration level 1 volumes should be larger than VTOCs on storage group volumes to accommodate the large number of small, compacted data sets. If you are using the DFSMSHsm small data set packing option, reduce your estimate for the number of data sets residing on the volume. Data set control blocks are not required for data sets stored in the small data set packing data set.

Any other volumes that might have a large number of small data sets, such as work or backup volumes, should also have larger VTOCs.

Converting Existing Volumes

You can use the DFSMSdss CONVERTV command to convert existing volumes to system-managed volumes without moving data from the volume. This is called *convert-in-place*. Because you must add the volume to a specific storage group, it is best to use CONVERTV on volumes that already contain primarily one type of data, for example, primary data.

Before actually converting volumes, simulate the process using the CONVERTV command with the SMS and TEST keywords. This simulation process determines whether a volume is eligible to become system-managed. A volume is considered ineligible when it contains data that cannot be system-managed. You must remove ineligible data sets from the volume before it can be completely converted. Also ensure that no data sets are allocated during the conversion, because data sets in use cannot be converted in place.

You can also use CONVERTV on volumes containing a mixture of data types. First, decide the storage group to which you want to add the volume. Then, move data off the volume that does not fit in the storage group, and then convert the volume. For more information on the CONVERTV command, see *DFSMS/MVS V1R2 DFSMSdss Storage Administration Guide*.

DASD volumes in a DFSMS environment are either system-managed or non-system-managed. However, when you use CONVERTV, there is an additional possibility that the volume might be left midway between system-managed and non-system-managed. This can happen if there are data sets on the volume you are converting that cannot be system-managed.

Indicators in the volume table of contents (VTOC) show whether a volume is fully converted to system management. These indicators represent the three ways that volumes are identified:

Converted

Indicates that the volume is system-managed. All data sets on the volume must have an assigned storage class and be cataloged in an integrated catalog.

Initial

Indicates that the volume is not ready for system management because it contains data sets that are either ineligible for system management or are not defined as system-managed.

During the convert-in-place process, system-managed and non-system-managed data sets can coexist on a volume that is in the initial state. However, this is an undesirable situation because you cannot allocate new data sets on a volume that is placed in the initial state. Always remove ineligible data sets from a volume before conversion. (See *MVS/ESA Storage Management Library: Managing Data* for more information on ineligible data sets.) If you do not clean up a volume before converting it, separate data ineligible for system management from data intended for system management as soon as possible after the conversion. Leaving volumes in an initial state for an extended time results in inefficient use of space.

You can separate the data by using the ISMF volume and data set list functions that provide filtering criteria. This helps to identify data that cannot be system-managed.

Non-System-Managed

The volume does not contain any system-managed data sets or it has not been initialized as system-managed.

Modifying the Volume Attribute List

The system mount and use attributes for storage group volumes are defined in the volume attribute list member (VATLSTxx) of SYS1.PARMLIB.

Specify volume attribute entries for all volumes to ensure that they are mounted with the appropriate mount and use attributes at IPL time.

Even though the use attribute is not used for system-managed volumes, specify a use attribute of *private* for system-managed volumes. This ensures the volume is private if you later change the volume to non-system-managed. You can specify a default use attribute to ensure that volumes not specifically defined in the volume attribute list are mounted with the private attribute. If a volume is not listed in the volume attribute list, and you do not have a default use attribute, the volume has a default of *public*. The volume is a candidate for system allocation of temporary data sets. The following is an example of specifying a default use attribute.

```
VATDEF IPLUSE(PRIVATE) SYSUSE(PRIVATE)
```

You can specify one entry with a generic volume serial number if you have a naming standard for volumes. For example, you could specify a list entry with a generic volume serial number of PRM* for all volumes starting with PRM.

You can have several volume attribute lists existing at the same time as long as each is stored in a different member. The VAL parameter in the IEASYS00 member of SYS1.PARMLIB indicates which volume attribute list the system should use.

IPLUSE applies to permanently resident volumes that are brought online during IPL (that is, they have no VATLST entry, or whose use attribute is not specified correctly in VATLSTxx).

SYSUSE applies to volumes that are varied online after IPL and have no entry in a VATLSTxx member.

Removing Volumes

As your storage needs grow, you will need to replace older DASD with newer models.

To remove a DASD volume:

1. Disable the volume for new allocations by changing SMS volume status to DISALL.
2. Remove all the data from the volume. For striped data sets, make sure that each stripe is removed from the storage group.
3. Vary the volume offline to MVS.
4. Delete the volume from the SMS configuration.

Once you have removed all the volumes contained in a DASD from the system, you can unplug and remove the device.

Defining a Pool Storage Group

You use ISMF to define storage groups. Figure 19 shows an example of defining a primary pool storage group. You can use the description field to describe the purpose of the storage group.

The storage group and management class for a data set are interrelated. For example, if you set AUTO MIGRATE or AUTO BACKUP to NO in the storage group definition, DFSMSHsm does not process any of the data sets in the storage group, regardless of the attributes in the management classes assigned to the data sets in the group.

After you define the attributes for the storage group, use ISMF to add volumes to the group. For more information on defining pool storage groups and adding volumes, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

```
                                POOL STORAGE GROUP DEFINE
COMMAND ===>

SCDS NAME:          STGADMIN.SMS.SCDS1
STORAGE GROUP NAME: PRIME90

TO DEFINE STORAGE GROUP, SPECIFY:
DESCRIPTION ===> Storage group for most permanent and temporary
                ===> data sets smaller than 285 MB (3390 DASD volumes)
AUTO MIGRATE  ===> Y (Y, N, I or P)  MIGRATE SYS/SYS GROUP NAME ===>
AUTO BACKUP   ===> Y (Y or N)        BACKUP SYS/SYS GROUP NAME ===>
AUTO DUMP     ===> Y (Y or N)        DUMP SYS/SYS GROUP NAME   ===>

DUMP CLASS   ===> ONSITE              (1 to 8 characters)
DUMP CLASS   ===> OFFSITE            DUMP CLASS   ===>
DUMP CLASS   ===>                   DUMP CLASS   ===>

ALLOCATION/MIGRATION THRESHOLD: HIGH ===> 95 (1-99)  LOW ===> 80 (0-99)
GUARANTEED BACKUP FREQUENCY         ===> 15       (1 to 9999 or NOLIMIT)

DEFINE SMS STORAGE GROUP STATUS     ===> N (Y or N)

USE ENTER TO PERFORM VERIFICATION AND SELECTION; USE HELP COMMAND
FOR HELP; USE END COMMAND TO SAVE AND EXIT; CANCEL TO EXIT.
```

Figure 19. An Example of Defining a Primary Pool Storage Group

Choosing Automatic Storage Group Processing Options

When you define a pool storage group, you decide whether DFSMSHsm should perform automatic space management, backup, and dump processing on the volumes within the system or system group. To do this, use the AUTO MIGRATE, AUTO BACKUP, and AUTO DUMP fields.

In a multihost DFSMS environment, you can use a system name or system group name to perform each of the DFSMSHsm automatic functions. If the name in the storage group is blank, any system that is part of the SMS complex can perform the processing. If you use a particular system name, only that system can perform DFSMSHsm automatic processing. If you use a system group name, any system that is part of the system group can perform the processing. A system is part of the system group if the system is defined as part of the sysplex with the same name, and not explicitly defined in the configuration.

Data sets are not actually backed up or migrated unless they are assigned a management class that specifies that they should be backed up, or that they can be migrated. The management class for the data set defines the backup and migration requirements.

You also need to specify the USERDATASETSERIALIZATION parameter on the DFSMSHsm SETSYS command to allow the movement of multivolume physical sequential data sets.

For more information about migration, backup, recall, and recovery, see *MVS/ESA SML: Managing Data*.

Using Guaranteed Backup Frequency

The GUARANTEED BACKUP FREQUENCY storage group attribute lets you specify the number of times within a defined period that you want data sets backed up. If you set AUTO BACKUP to YES, this attribute is required, otherwise it is optional.

You can specify the maximum number of days that can elapse between backups, from 1 to 9999. You can also set NOLIMIT, which will back up data sets according to management class specifications. There is no default.

When a data set is backed up due to change activity, and the maximum number of backup copies to be kept of the data set has been reached, DFSMSHsm deletes the oldest backup version of the data set. In contrast, guaranteed backup frequency backups taken when the change indicator in the VTOC is off only replace the most recent backup version of the data set. For these data sets, the oldest backup version is **not** marked for deletion.

Nothing identifies guaranteed backup frequency backups as different from incremental backups made because the data set was changed.

The benefit of using guaranteed backup frequency is that **all** data sets that exist on DFSMSHsm-managed primary volumes are automatically backed up within a specified time period. However, the disadvantages are:

- More incremental backups are created
- The need to recycle backup tape volumes increases.

You can increase the speed of volume recovery by having a volume dump from which to restore much of the volume. This reduces the maximum number of incremental backup tapes that might be required to recover the volume to the latest available backup.

Setting Thresholds for Storage Groups

When you define a pool storage group, specify values for the high and low thresholds for the volumes within the storage group. The *thresholds* define an upper and lower space limit for the volumes in a pool storage group, and are used as aides in selecting volumes for allocation and data migration.

Because thresholds are defined as percentages and not absolute values, pool storage groups can contain volumes with identical device geometries but different capacities without creating problems.

For example, you can combine all models of the 3390 device into one pool storage group. Because the 3390 models have different capacities, the threshold levels you set indicate different absolute space values for each device model. You should keep this in mind when you define storage groups used for large data set allocations and ensure the smallest capacity device in the group contains sufficient free space.

Setting the High Threshold

The *high threshold* is used to determine candidate volumes for new data set allocations. The system tries not to exceed the high threshold. It is also used during interval migration to determine if data sets should be migrated off the volume.

The space remaining between the high threshold and the volume's capacity is available for new secondary extents for the data sets residing on the volume. Thus, the value you use for the high threshold should supply adequate space for data set growth between space management cycles. If the high threshold is exceeded, data is migrated during the next space management cycle until the low threshold is reached or the remaining data sets on the volume must remain there due to the requirements set in the assigned management classes.

One way to determine the high threshold value is to pick a percentage without considering the amount of physical space that value will leave as free space on the volume. Then, use that value without regard for the capacity of the volumes. Thus, the larger capacity volumes have more free space available for secondary allocations, which might be appropriate because the volumes contain more data than the smaller capacity volumes.

Figure 20 shows an example of setting a high threshold value of 90% for all volumes, and the amount of space that is left available on various volumes.

Figure 20. Free Space Available when using a 90% Threshold Value

DASD Model	Free Space (in MB)
3380 Models D, J	63
3380 Model E	126
3380 Model K	189
3390 Model 1	95
3390 Model 2	189
3390 Model 3	284
3390 Model 9	852
9345 Model 1	100
9345 Model 2	150

Conversely, you can determine the high and low threshold values you want to use by determining how many MBs of free space you want to keep on your volumes. This allows you to define higher threshold values for your larger capacity devices while maintaining adequate free space. If you use spill storage groups, as explained in “Using Storage Overflow to Prevent Allocation Failures” on page 54, you can set even higher threshold values without risking allocation failures.

Figure 21 on page 52 shows an example of calculating the high threshold value to guarantee a minimum of 63 MB free space for volumes of different capacities.

In general, set the high threshold value lower for the large storage group than the primary storage group to allow for larger secondary extents.

Setting the Low Threshold

The *low threshold* is used during primary space management to determine the amount of free space to provide on the volume.

Set your low threshold value according to how much space is needed between space management cycles for data processing. The difference between the high and low threshold values is the space that is available for new allocations. The system tries not to allocate over the threshold, but will do so if no other volume is available.

Determine the low threshold value, based on the space required in the storage group, not on the space required on a given volume. Adjust the threshold value as you replace volumes with volumes that have a different capacity.

For example, Figure 21 on page 52 shows the number of volumes required for 50 GB of storage for various DASD models. If you use the 3380 Model J as a basis for comparison, you see that a low threshold of 80% coupled with a high threshold of 90% provides 5040 MB of space for new allocations across the 80 volumes. However, if you replace the 3380 Model J volumes with 27 3390 Model 2 volumes, and adjust the high threshold to 97%, you have to set the low threshold at 87% to provide the minimum 5040 MB of space.

Figure 21. Example of Setting Thresholds Based on Required MBs

DASD Model	Number of Volumes for 50 GB	High Threshold		Low Threshold		Space (MB) Available for New Allocations
		%	Free MB	%	Free MB	
3380 Model J	80	90	63	80	126	5040
3380 Model E	40	95	63	85	189	5040
3380 Model K	27	96	75	86	264	5103
3390 Model 1	53	93	66	82	170	5512
3390 Model 2	27	96	75	86	264	5103
3390 Model 3	18	97	85	87	368	5094
3390 Model 9	6	99	85	89	936	5106
9345 Model 1	50	93	70	82	180	5500
9345 Model 2	34	95	75	85	225	5100

Note: In this example, the installation requires a minimum 5040 MB of space to be available for new allocations between space management cycles. The example is based on maintaining 50 GB of space in the storage group, regardless of the DASD model used. The values for high and low threshold are expressed in percentages (%), and also in the number of MB the threshold value leaves available on the volume. The high threshold value is chosen to maintain a minimum of 63 MB of free space on the volume for new secondary extents. The value for space available for new allocations is determined by subtracting the low threshold MB value from the high threshold MB value, and multiplying the result by the number of volumes.

The numbers in Figure 21 assume that you do not combine models with different capacities in the same storage group. However, you can mix capacities within a storage group, even though you should never mix devices with different geometries. If you mix volumes with different capacities, you need to make the calculations for high and low threshold values, based on the combined number of bytes available within the storage group. Figure 22 on page 53 shows an example of calculating threshold values for storage groups containing a mixture of 3390 models.

Example A

Number of Volumes	Device Type	Total Capacity
9	3390 Model 1	8514 MB
10	3390 Model 2	18920 MB
8	3390 Model 3	22704 MB
Totals: 27		50138 MB

Space needed = 5040 MB: $\frac{5040}{50138} = .101$, rounds to 11%

High Threshold, 63 MB per volume: $\frac{63 \times 27}{50138} = .034$, rounds to 4%

Result: High Threshold = 96%
Low Threshold = 85%

Example B

Number of Volumes	Device Type	Total Capacity
21	3390 Model 1	19866 MB
10	3390 Model 2	18920 MB
4	3390 Model 3	11352 MB
Totals: 35		50138 MB

Space needed = 5040 MB: $\frac{5040}{50138} = .101$, rounds to 11%

High Threshold, 63 MB per volume: $\frac{63 \times 35}{50138} = .044$, rounds to 5%

Result: High Threshold = 95%
Low Threshold = 84%

Example C

Number of Volumes	Device Type	Total Capacity
41	3390 Model 1	38786 MB
6	3390 Model 2	11352 MB
0	3390 Model 3	0 MB
Totals: 47		50138 MB

Space needed = 5040 MB: $\frac{5040}{50138} = .101$, rounds to 11%

High Threshold, 63 MB per volume: $\frac{63 \times 47}{50138} = .059$, rounds to 6%

Result: High Threshold = 94%
Low Threshold = 83%

Figure 22. Calculating Thresholds for Storage Groups with Mixed Capacity Volumes. Values must be rounded up because the space requirements are minimum, not approximate, values.

Set the threshold values lower for the large storage group, and the database storage groups if you have large databases, than for the primary storage group. The high threshold needs to be lower to allow space for large secondary extents, and the low threshold needs to be lower to ensure adequate space for new large data set allocations.

Specifying Storage Group and Volume Connectivity

You use the ISMF Storage Group Application to define the status of a storage group and the volumes within the storage group to the systems or system groups that have access to it. The status indicates whether or not the storage group is connected to a system or system group and the type of access that is allowed.

The values you can use are described in *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

A status of NOTCON indicates that the storage group or volumes are not physically connected, which has the same effect as being undefined to the system or system group. The storage group status overrides the status of individual volumes within the storage group: if you specify a storage group to a system or system group with the status of NOTCON, the volumes in the storage group are not selected by that system or system group regardless of the status of individual volumes. To avoid confusion, you should ensure that the system or system group connectivity to a storage group and to the volumes within the storage group is the same.

If you use any value other than NOTCON for a system or system group (ENABLE, DISABLE, DISNEW, QUIALL, or QUINEW), SMS assumes that the storage group is connected to the system or system group. Ensure that the system or system group has physical access to all devices in the storage group.

You can dynamically change any system or system group status, except the NOTCON status. If you want a system or system group to access the volumes belonging to the storage group, first ensure that the volumes have a physical connection to the system or system group, then activate a new configuration that defines the connection.

Using Storage Overflow to Prevent Allocation Failures

New data sets are not usually allocated on a volume that has met or exceeded its high threshold. To prevent an allocation failure, you can provide overflow or spill storage at the volume or storage group level. This lets you set the high threshold at a higher level in the primary storage groups, and fully use the storage capacity available.

You can direct system-managed data sets to either a spill volume or storage group by taking advantage of the volume or storage group SMS quiesce status. Depending on your procedure, the overflow or spill storage is provided at the volume or storage group level.

To implement storage overflow, set the SMS status to QUIALL or QUINEW for the overflow volumes or storage groups. You can designate single volumes as overflow storage within the current pool storage groups to allow each storage group pool to have its own overflow volumes. Or, you can quiesce an entire pool storage group to become a common overflow for multiple storage groups. In the latter, you must include the overflow storage group as one of the eligible storage groups selected by your storage group ACS routine. However, this only works in a JES2 environment; in a JES3 environment, it could prevent jobs from being scheduled.

To spill data to another volume, direct the allocation to an SMS ENABLED storage group that contains both SMS enabled and quiesced volumes. All allocations are directed to the volumes in ENABLE status as long as the space usage for these volumes is below the high threshold. When all these volumes exceed the high threshold, or cannot satisfy a new allocation or extent, the volume in QUIALL or QUINEW that is the closest match to the storage class objectives is selected. This causes all the volumes in ENABLE status to be filled to the threshold limit. If a new data set allocation exceeds the high threshold of all available enabled volumes, the allocation spills to the volumes in QUIALL or QUINEW status.

To spill data to another storage group, direct the allocation to two storage groups. Set one storage group to an SMS status of ENABLE and the other to an SMS status of QUIALL or QUINEW. The allocations are satisfied by the storage group in ENABLE status until filled to the threshold limit. If a new data set allocation exceeds the high threshold of all enabled volumes, the allocation spills to the volumes in the storage group with an SMS status of QUIALL or QUINEW. A volume is quiesced if either the storage group or the volume is quiesced. SMS treats quiesced volumes and quiesced storage group volumes equally.

When spilling to overflow volumes, SMS tries to fill the quiesced volumes evenly, possibly not allowing enough space for very large data sets. To reserve space for the large data sets, you can use two tiers of quiesced volumes.

Set a high threshold for the volumes in the first tier. You can only set the threshold at a storage group level. For example, the first tier would consist of quiesced overflow volumes or storage groups that contain larger high threshold values than the second tier storage groups. The high threshold of the first tier might be the same as the enabled storage group. For more information on setting thresholds, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

A second tier would consist of a quiesced storage group containing a high threshold setting that can easily accommodate large data sets. If the typical large data set occupies 70% of a volume in the quiesced storage group, you might set the high threshold to 30%, reserving the final 70% for large data sets. SMS first selects volumes that can satisfy the allocation request without exceeding the high threshold. In effect, second tier volumes will be placed at the end of the selection list earlier than first tier volumes.

If you implement the two tier technique, you might not want to use interval migration for the storage group. To turn off interval migration for the tier two storage group, enter a P for the storage group AUTO MIGRATE attribute. For more information on storage group attributes, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

Managing Data with Pool Storage Groups

In this section, we provide you with sample storage group definitions and ACS routines to migrate and manage interactive, large, batch, temporary, database, and object data in a DFSMS environment. These definitions and the routine are for guidance. You might need to change them to meet your needs.

As you create your definitions and routines, consider:

- Changing volume serial numbers to match your volume naming conventions
- Evaluating the size of your large data sets and changing the comparison value in the storage group ACS routine
- Using your naming conventions to customize filtering for the ACS routine.

Sample Pool Storage Groups

The storage groups to be assigned by our sample storage group ACS routine are shown in Figure 23 on page 57. Following is a description of the storage groups used in the ACS routine:

PRIME80, PRIME90

The primary storage groups, assigned to most system-managed data. This data includes interactive, batch, VSAM, and multivolume data sets and some temporary data. These storage groups are also assigned to non-production database data.

Only data sets whose maximum size is 285 MB or smaller are assigned to these storage groups.

PRIME80 contains 3380 DASD volumes of all models. PRIME90 contains 3390 DASD volumes of all models. Both storage groups allow automatic migration, backup, and dump processing.

LARGE80, LARGE90

The large storage group, assigned to data sets larger than 285 MB. This includes database image copies, archive logs, some temporary data sets, and large interactive, batch, VSAM, striped sequential, and multivolume data sets.

LARGE80 contains 3380 DASD volumes of all models. LARGE90 contains 3390 DASD volumes of all models. Both storage groups allow automatic migration, backup, and dump processing.

The high and low migration thresholds are set at 75% and 60%, respectively, so that sufficient space is available for new and extended data sets.

VIO

The VIO storage group, assigned to temporary non-VSAM data sets smaller than 20 MB, except DFSORT temporary data sets, which are assigned to the primary or large storage groups. For more information about preventing DFSORT temporary data from using the VIO storage group, see “Writing a Storage Class ACS Routine for Temporary Data Sets” on page 59.

DB2

The DB2 database storage group, assigned to DB2 production database data. This storage group can contain either 3380 or 3390 DASD volumes, but not both. It allows automatic migration, backup, and dump processing.

CICS

The CICS database storage group, assigned to CICS production database data. This storage group can contain either 3380 or 3390 DASD volumes, but not both. It allows automatic migration, backup, and dump processing.

IMS

The IMS database storage group, assigned to IMS production database data. This storage group can contain either 3380 or 3390 DASD volumes, but not both. It allows automatic migration, backup, and dump processing.

Figure 23. Sample DASD Storage Groups

Attributes	Primary and Large Storage Groups			
NAME	PRIME80	PRIME90	LARGE80	LARGE90
TYPE	POOL	POOL	POOL	POOL
AUTO MIGRATE	YES	YES	YES	YES
AUTO BACKUP	YES	YES	YES	YES
AUTO DUMP	YES	YES	YES	YES
DUMP CLASS	ONSITE, OFFSITE	ONSITE, OFFSITE	ONSITE, OFFSITE	ONSITE, OFFSITE
HIGH THRESHOLD	95	95	75	75
LOW THRESHOLD	80	80	60	60
GUARANTEED BACKUP FREQUENCY	15	15	15	15
Attributes	Database and VIO Storage Groups			
NAME	CICS	DB2	IMS	VIO
TYPE	POOL	POOL	POOL	VIO
VIOMAXSIZE	—	—	—	20 MB
VIO UNIT	—	—	—	3380
AUTO MIGRATE	YES	YES	YES	—
AUTO BACKUP	YES	YES	YES	—
AUTO DUMP	YES	YES	YES	—
DUMP CLASS	ONSITE, OFFSITE	ONSITE, OFFSITE	ONSITE, OFFSITE	—
HIGH THRESHOLD	75	75	75	—
LOW THRESHOLD	60	60	60	—
GUARANTEED BACKUP FREQUENCY	NOLIMIT	NOLIMIT	NOLIMIT	NOLIMIT

Writing the Storage Group ACS Routine for DASD Data

General-use programming interface

The storage group ACS routine fragment shown in Figure 24 assigns the storage groups shown in Figure 23 on page 57. See *MVS/ESA SML: Implementing System-Managed Storage* for an example of the complete storage group ACS routine.

This routine uses filter lists to identify production data used with CICS, DB2, and IMS. These data sets are assigned to the CICS, DB2, or IMS storage groups, depending on the database product used.

Large temporary data sets are placed in the large storage groups LARGE80 or LARGE90.

Temporary DASD data sets are assigned to the VIO storage group if they are smaller than the VIOMAXSIZE attribute defined for the storage group (in our sample group, 20 MB) and the data set is assigned the STANDARD storage class. This prevents temporary data sets assigned the NONVIO storage class from using VIO. This is done so that DFSORT data sets are assigned to the primary or large storage groups rather than the VIO storage group, because these data sets should not use VIO. See Figure 25 on page 60 for an example of identifying VSAM and DFSORT temporary data sets, and assigning them an appropriate storage class.

```

/*****
/* Start of FILTLIST Statements */
/*****
FILTLIST CICS      INCLUDE(PCICS*,**)
                  EXCLUDE(**.DFHXRMMSG,**.DFHXRCTL,**.LOADLIB,
                        **.COB2CICS,**.COB2LIB,**.PLILINK,
                        **.DFHJ01%,**.DFHRSD,**.DFHCSD,
                        **.DFHINTRA,**.DFHTEMP)
FILTLIST DB2      INCLUDE(PDB*,**)
                  EXCLUDE(**.DIRECTRY,**.LOG*)
FILTLIST IMS      INCLUDE(PIMS*,**)
                  EXCLUDE(**.LGMSG,**.OL*,**.QBLKS,**.RDS,**.SHMSG,
                        **.SPA,**.WADS*,**.RECON)
/*****
/* End of FILTLIST Statements */
/*****
/*****
/* Start of Mainline SELECT */
/*****
SELECT
  WHEN (&DATACLAS = 'TMMTEMP')          /* Temporary tape data sets*/
  DO                                     /* go to the LARGE group */
    SET &STORGRP = 'LARGE90','LARGE80'
    EXIT
  END
  WHEN (&DSTYPE = 'TEMP' &&            /* Small temporary data */
        &STORCLAS = 'STANDARD' &&     /* are directed to VIO or */
        &MAXSIZE < 285MB)            /* the PRIME storage group */
  DO
    SET &STORGRP = 'VIO','PRIME90','PRIME80'
    EXIT
  END
END

```

Figure 24 (Part 1 of 2). Sample Storage Group ACS Routine for DASD Data Sets

```

WHEN (&DSN = &CICS)                /* Isolate CICS databases */
DO
    SET &STORGRP = 'CICS'
    EXIT
END
WHEN (&DSN = &DB2)                  /* Isolate DB2 databases */
DO
    SET &STORGRP = 'DB2'
    EXIT
END
WHEN (&DSN = &IMS)                  /* Isolate IMS databases */
DO
    SET &STORGRP = 'IMS'
    EXIT
END
WHEN (&MAXSIZE => 285MB)            /* Large data sets go to */
DO                                  /* their own storage group */
    SET &STORGRP = 'LARGE90','LARGE80'
    EXIT
END
OTHERWISE                            /* Normal and medium-sized */
DO                                  /* data sets go to the */
    SET &STORGRP = 'PRIME80','PRIME90' /* PRIME storage group */
    EXIT
END

END
/*****
/* End of Mainline SELECT */
*****/

```

Figure 24 (Part 2 of 2). Sample Storage Group ACS Routine for DASD Data Sets

All other data sets are placed in the primary or large storage groups, based on the maximum size of the data set. Data sets smaller than 285 MB are placed in PRIME80 or PRIME90, those 285 MB or larger are placed in LARGE80 or LARGE90.

Writing a Storage Class ACS Routine for Temporary Data Sets

You can allocate most temporary data sets as VIO data sets to improve performance. However, we do not recommend that you allocate DFSORT temporary output and work data sets as VIO data sets. VIO reduces elapsed time at the cost of increased CPU time for DFSORT applications.

You should allocate DFSORT temporary data sets as VIO data sets only under the following circumstances:

- Elapsed time is of greater concern than CPU time, or
- The temporary data sets are passed to or from other job steps whose performance can be improved by the use of VIO.

To ensure that all temporary data sets are system-managed effectively, design your storage class and storage group ACS routines as follows:

1. Write the storage class ACS routine so that all temporary DFSORT data sets are assigned to a storage class you can identify as requiring non-VIO storage. Use the &DD variable to identify data sets assigned ddnames used by DFSORT. You need to do this in the storage class routine because you cannot use the &DD variable in the storage group routine.

Figure 25 is a fragment of a storage class ACS routine that assigns DFSORT temporary data sets to a non-VIO storage class. For an example of the complete storage class ACS routine, see *MVS/ESA SML: Implementing System-Managed Storage*.

```

FILTLIST DFSORTDD          INCLUDE(SORTWK*,SORTDK*,SORTOUT)

:
SELECT

:
WHEN (&DSTYPE = 'TEMP' &&          /* Do not direct DFSORT */
      &DD = &DFSORTDD)             /* temp data sets to VIO */
DO
  SET &STORCLAS = 'NONVIO'
  EXIT
END

:
END

```

Figure 25. Assigning Temporary DFSORT Data Sets to a non-VIO Storage Class

2. Write the storage group ACS routine to assign temporary data sets to either VIO or non-VIO storage groups, based on the data set's size and storage class.

Figure 26 is a fragment of a storage group ACS routine that assigns data sets to storage groups by selecting on the STANDARD storage class. Because the temporary DFSORT data sets are assigned the NONVIO storage class, they get allocated to the primary or large storage groups rather than the VIO storage group.

```

SELECT

:
WHEN (&DSTYPE = 'TEMP' &&          /* Small temporary data */
      &STORCLAS = 'STANDARD' &&    /* are directed to VIO or */
      &MAXSIZE < 285MB)           /* the PRIME storage group */
DO
  SET &STORGRP = 'VIO','PRIME90','PRIME80'
  EXIT
END

:
WHEN (&MAXSIZE => 285MB)           /* Large data sets go to */
DO                                  /* their own storage group */
  SET &STORGRP = 'LARGE90','LARGE80'
  EXIT
END
OTHERWISE                           /* Normal and medium-sized */
DO                                  /* data sets go to the */
  SET &STORGRP = 'PRIME80','PRIME90' /* PRIME storage group */
  EXIT
END

END

```

Figure 26. Directing Temporary Data Sets to Appropriate Storage Groups: Example 1

Figure 27 shows an alternate approach where you explicitly check for the NONVIO storage class. All temporary data sets 285 MB or larger, whether assigned the STANDARD or NONVIO storage class, are assigned to the large storage groups.

```
SELECT
:
WHEN (&DSTYPE = 'TEMP' &&          /* Small temporary data */
      &STORCLAS = 'STANDARD' &&    /* are directed to VIO or */
      &MAXSIZE < 285MB)           /* the PRIME storage group */
DO
  SET &STORGRP = 'VIO','PRIME90','PRIME80'
  EXIT
END
WHEN (&STORCLAS = NONVIO &&        /* Small and medium size */
      &MAXSIZE < 285MB)           /* temporary data sets for */
DO                                  /* non-VIO storage */
  SET &STORGRP = 'PRIME90','PRIME80'
  EXIT
END

:
WHEN (&MAXSIZE => 285MB)           /* Large data sets go to */
DO                                  /* their own storage group */
  SET &STORGRP = 'LARGE90','LARGE80'
  EXIT
END

:
END
```

Figure 27. Directing Temporary Data Sets to Appropriate Storage Groups: Example 2

End of General-use programming interface

Chapter 4. Maintaining Pool Storage Groups

The structure of storage groups you have developed might be well suited to your current environment. However, chances are that your current environment will change, and the structure of the storage groups that you have developed will also need to change. The exact mixture of data that you manage evolves; data set size and number grow, and new devices are installed. You might need to modify existing storage groups or add new ones to meet changing data processing requirements, and to grow along with the storage management tools to expand the scope of system-managed storage.

Adding or Deleting Storage Groups or Devices

You might find that as your business environment matures, the types of data you handle also change. For example, in most organizations, applications are becoming more interactive, and databases are expanding. Steady changes in data types and business procedures can eventually change some of your requirements for specific storage groups. You might:

- Move ISAM and unmovable data sets into the primary storage group when they have been converted to a manageable format.
- Add new volumes to the primary storage group as you move more data into it.
- Delete a storage group that is no longer needed because the data that was in the storage group has been moved to the primary storage group.

You use the ISMF Storage Group Application to add or delete DASD volumes. Keep the following in mind when modifying storage group definitions:

- When you make a change to the storage group definition, ensure that the volume-to-processor connectivity is consistent with the volume status in the storage group definition.
- If you add DASD volumes to a storage group to improve performance, to make the most of availability, or increase capacity, ensure that your storage class definitions reflect the capabilities of your hardware.
- If you move data between storage groups, ensure that the management class and storage class requirements of the data can still be met. Ensure that the DFSMSHsm processing requirements specified in the management class for the data set are provided for the new storage group, and that the performance and availability requirements specified in the storage class can be met.
- When you delete a DASD volume from a storage group, ensure that all the data sets and objects, stored in DB2 table spaces, are first moved off the volume. Check for multivolume data sets on the volume you are removing, and move them. If the volume is no longer managed by the system, you need to change the CONVERTED status in the VTOC. You can do this by using the DFSMSdss CONVERTV NONSMS command. You can also reinitializing the volume by using the ICKDSF INIT command without the STGR parameter.

Monitoring Pool Storage Groups

Set reasonable goals for performance, data availability, and space usage. This ensures that SMS is providing service that is both predictable and acceptable. These goals are derived from user data requirements and published in the form of a *service-level agreement* between the storage administration group and the user groups. *MVS/ESA SML: Managing Data* describes in detail how to determine service levels and how to negotiate and prepare service-level agreements.

Develop procedures for managing the storage groups and measuring the level of service being delivered for the data after setting reasonable goals. This section describes how to monitor performance. The following section discusses how to manage availability and space across the storage groups.

One of the keys to successful storage management is to automate or streamline as many of the routine storage management tasks as possible, leaving the people in your organization free to be more productive in their own job assignments. Toward this end, use the space management and reporting functions of ISMF, DFSMSHsm, DFSMSdss, the IDCAMS DCOLLECT command, Resource Measurement Facility (RMF) and Service Level Reporter (SLR), to manage and monitor storage groups. Avoid procedures that require excessive manual involvement, operator intervention, or user resources whenever possible.

Using the DCOLLECT Command

The DFSMS data collection facility, DCOLLECT, is recommended for:

- Capacity planning for DASD volumes and DFSMSHsm-owned tape volumes
- Storage and space administration measurement and reporting
- Service-level reporting
- Cost accounting.

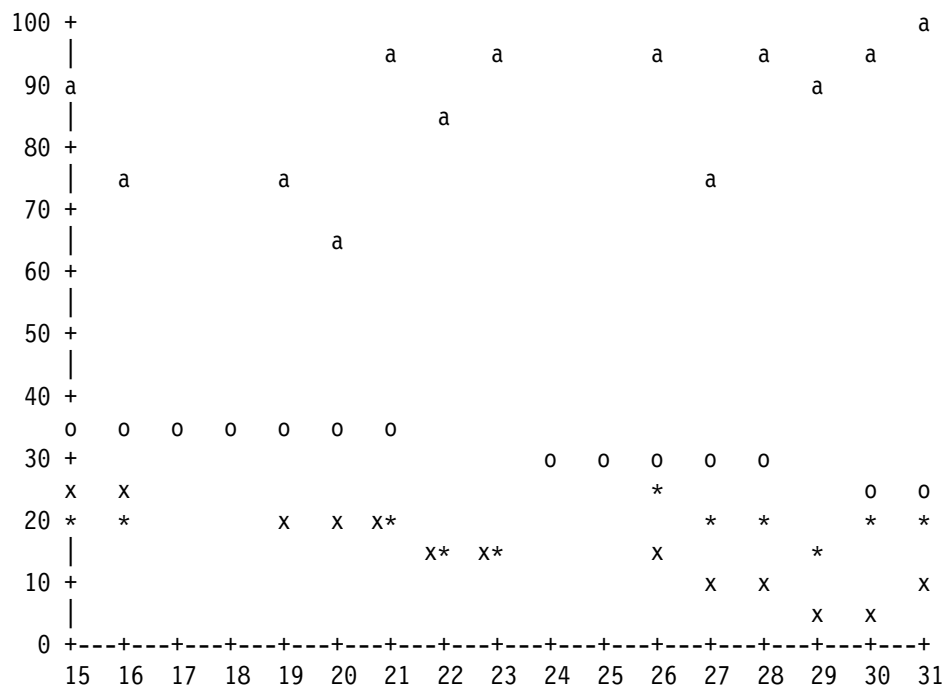
The examples below demonstrate how output can be generated for storage usage measurement and reporting, and capacity planning.

Pool Storage Group Measurement and Reporting

The graph in Figure 28 on page 65 shows an example of free space trend reporting for each storage group on a daily basis. This report is useful to the storage administrator, capacity planner, and service analyst to help them identify:

- If the daily free space available in the storage groups is meeting the service requirements documented in the service-level agreement.
- If the storage group thresholds are correctly set for a combination of:
 - The number and capacity of volumes contained in each storage group
 - The number of data sets allocated to each storage group
 - The data set age before migration or deletion.
- If the total capacity is sufficient for each storage group's usage.

The graph is produced from data contained in the DCVPERCT field of the DCOLLECT volume record.



LEGEND:

Vertical axis : Average % free space

Horizontal axis : Days in the month

* : PRIME storage group

x : LARGE storage group

o : Non-system managed storage (private volumes)

a : Scratch pool (non-system managed, pooled volumes)

Figure 28. Report on Average Free Space for Each Storage Group

From the information provided by the SLR report, you can see where you might need to add volumes to a storage group.

You can use this report to identify whether the thresholds for a storage group need changing, or whether more volumes need to be added to a storage group. You can also use the information to evaluate whether the migration ages in the management classes need changing.

The service analyst can use this report to identify if the free space available in each storage group is matching the free space levels that are documented in the service-level agreements.

Capacity Planning and Trend Reporting

The report in Figure 29 shows the installed versus allocated DASD capacity. You can use this report to help identify:

- Your DASD growth versus usage rates
- Whether DASD usage is increasing or decreasing at an expected amount
- Whether installed DASD capacity can meet planned use.

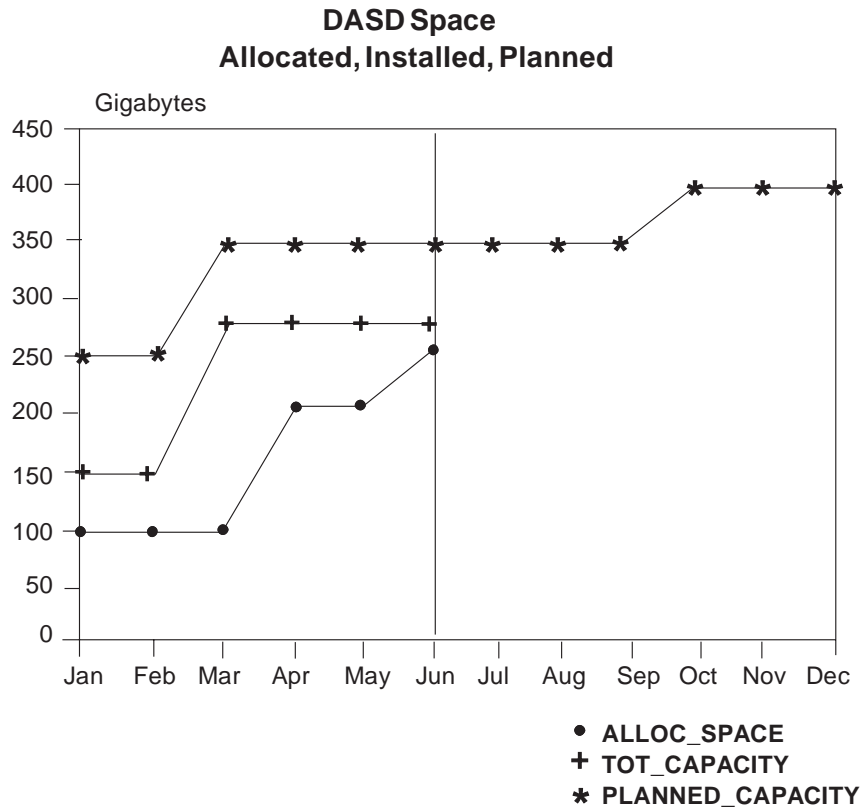


Figure 29. DASD Capacity - Allocated, Installed, and Planned

From the graph, you can see that:

- The planned capacity is greater than the installed capacity. This could be attributable to several reasons:
 - Volumes can be installed but offline
 - Some volume problems can be present
 - DASD is not being installed in a timely fashion
 - Planned capacity requirements are greater than actually needed.
- The installation increased its installed DASD capacity by 86%, but the rate of allocated space increased by 150%.

The two sample reports are generated by SLR, using data collected using the access method services DCOLLECT command. You can specify the data, the basis of the SLR reports, as shown in the sample DCOLLECT data entry panels in Figure 30 on page 67.

For information on using ISMF for DCOLLECT, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*. For more information on using the IDCAMS DCOLLECT command, see *DFSMS/MVS V1R2 Access Method Services for ICF*.

```
DATA COLLECTION ENTRY PANEL                Page 1 of 2
COMMAND ==>>

SELECT DATA COLLECTION OPTIONS:
  DATA SET INFORMATION   ==>> N      (Y or N; Y requires volume(s) or
  VOLUME INFORMATION     ==>> Y      storage group(s) on next page)
  MIGRATION DATA        ==>> Y      (Y or N)
  BACKUP DATA           ==>> Y      (Y or N)
  CAPACITY PLANNING DATA ==>> Y      (Y or N)

SPECIFY OUTPUT DATA SET:                   (1 to 44 characters)
  DATA SET NAME         ==>> DCOLLECT.CAPLAN
  OPTIONAL PASSWORD      ==>>         (Ignored if SMS-managed data set)
  REPLACE CONTENTS       ==>> N      (Y or N)
  NUMBER OF DATA SETS   ==>> 10000   (1 to 99999999; new data set only)

SPECIFY INPUT DATA SET:                   (1 to 44 characters)
  MIGRATION DATA SET NAME ==>> 'HSM.MCDS'
  BACKUP DATA SET NAME   ==>> 'HSM.BCDS'

USE ENTER TO PERFORM SELECTION; USE DOWN COMMAND TO VIEW NEXT ENTRY PANEL;
USE HELP COMMAND FOR HELP; USE END COMMAND TO EXIT.
```

Figure 30 (Part 1 of 2). Using DCOLLECT through ISMF

```

                                DATA COLLECTION ENTRY PANEL                                Page 2 of 2
COMMAND ==>

SPECIFY POOL STORAGE GROUPS (full name)
====> PRIME                                ====>
====> LARGE                                ====>
====>                                     ====>
====>                                     ====>
====>                                     ====>

SPECIFY VOLUMES (full, partial, *****, or *)
====> SCR*                                ====>      ====>      ====>
====> PRIV*                               ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>
====>                                     ====>      ====>      ====>

USE ENTER TO PERFORM SELECTION; USE UP COMMAND TO VIEW PREVIOUS ENTRY PANEL;
USE HELP COMMAND FOR HELP; USE END COMMAND TO EXIT.

```

Figure 30 (Part 2 of 2). Using DCOLLECT through ISMF

Using SLR and RMF Reports to Verify Performance

From a user's perspective, performance is measured by how long it takes the system to respond to a request for input or output. In fact, good performance is measured not only in fast response time for a request, but in predictable and timely job completion.

Performance depends on many factors, such as the amount and rate of I/O activity against volumes and their capacity, and the hardware configuration, including device types and channel paths. Managing performance means balancing system resources, such as processor time, channel use, and I/O rate, according to user requirements.

You can measure performance for individual volumes or for storage groups by using the SLR and RMF licensed programs to record and report statistics by system, subsystem, and device. SLR uses RMF output to measure TSO response time, the number of concurrent TSO users, workload size, and device activity. RMF measures service in terms of transaction rates, device contention, and queuing in the device or channel path. Reports by device, logical control unit, and channel path can help you measure performance across the system. You can also produce reports by storage group. For more information and examples of these reports, see *MVS/ESA SML: Managing Data*.

You can report on cache performance using the Cache RMF Reporter program. The program runs as an exit to RMF and writes cache statistics to the SMF data set.

RMF can also produce reports that describe the jobs delayed while waiting for service from DFSMSHsm. The DFSMSHsm delay report identifies the DFSMSHsm function code that explains why DFSMSHsm has delayed each job. This report can help you determine when DFSMSHsm activity is adversely impacting the performance of your system.

RMF and SLR reports should be run at regular intervals during the day, based on time periods set in the SMFPRMxx and ERBRMFxx members of SYS1.PARMLIB. Reports are typically run every 15 to 60 minutes, depending on the level of system activity.

Identifying Potential Problems

Compare the performance measurements in the reports with the targets you have set to see whether you are meeting your goals for service. Save measurements over a period of time and analyze the trend of the values to identify potential problems before they occur.

You can also use RMF reports to identify a specific performance problem such as a heavily contended volume or channel path. You can start to determine how hard the system is working to meet your service goals by isolating the cause of response time problems.

You might need to improve performance if the performance statistics you measure consistently exceed the targets specified in your service-level agreement, or if the system is working harder than you had anticipated to meet your targets. For example, you might solve a performance problem by using storage class to place high-contention data on a cached volume; by reconfiguring strings to correct an I/O bottleneck; by isolating system paging and swapping; or by adding more DASD capacity.

The Storage Subsystem Library describes the kinds of actions you can take to improve performance in your hardware configuration.

For more information on using RMF to produce reports, see *RMF Monitor I and II Reference and User's Guide* and *RMF Monitor III Reference and User's Guide*. For details on using SLR to supplement RMF statistics, see *SLR User's Guide: Reporting Planned Availability*.

Managing Availability for Pool Storage Groups

Availability in a computing system is defined as the degree to which a resource is ready when needed to deliver service to users. Although users are concerned primarily with availability of data, their perceptions of availability might include problems with hardware, software, networks, subsystems, and applications.

At the physical or hardware level, availability is ensured through configuration—backup equipment, channel and path switching, and maintenance for critical components.

The Storage Subsystem Library describes how to improve hardware availability. At the logical or data set level, availability measures include immediate recovery using dual copy, regular data set backup, journaling, sharing, and disaster recovery procedures.

Although some data sets might have specialized availability requirements, manage data availability primarily by management class and storage class (for hardware duplexing) as detailed in your service-level agreement.

Backing up and Recovering DASD Volumes

Recovering an entire volume from data set copies can be a time-consuming process. However, you might find that it is not feasible to back up entire volumes daily because volume dumps can tie up system resources for long periods. Consider supplementing daily data set backups with periodic volume dumps.

You can increase the speed of volume recovery by having a volume dump, which enables you to restore much of the volume. This reduces the maximum number of incremental backup tapes that might be required to forward recover the volume to the latest available backup.

Specifying AUTO DUMP in the storage group definition provides automatic full volume backup for your storage group volumes. DFSMShsm calls DFSMSdss full volume dump to perform the backup. You can automatically back up all of the volumes in your storage groups this way.

The DFSMShsm invocation of DFSMSdss full volume dump function provides automatic record keeping in a data set, eliminating the need to maintain DFSMSdss output information. It can also replace JCL or application programs written to call DFSMSdss. For more information on using DFSMShsm invocation of DFSMSdss full volume dump, see *DFSMS/MVS V1R2 DFSMShsm Storage Administration Guide*.

Data set backup, as specified through management class, is provided by the incremental backup function of DFSMShsm. You can provide for both full volume restore and incremental volume recovery by using the full volume dump and incremental backup functions of DFSMShsm. You can use both functions to provide a more accurate reconstruction of a volume's contents and make the recovery faster. It is more accurate than only full volume dumping, but faster than using incremental backup alone.

Using DFSMShsm for Automatic Incremental Backup

You can provide effective and efficient data recovery for both system-managed and non-system-managed storage by using DFSMShsm automatic incremental backup and full volume dump invocation of DFSMSdss. Following is an example of how to define DFSMShsm commands to provide automated control of backup processes.

```
SETSYS BACKUP
SETSYS AUTOBACKUPSTART(0030 0330 0530)
SETSYS AUTODUMPSTART(0030 0330 0530)
SETSYS DUMPIO(3)
SETSYS MAXDUMPTASKS(3) MAXBACKUPTASKS(3)
SETSYS TAPEDELETION(HSMTAPE) SELECTVOLUME(SPECIFIC)

DEFINE BACKUP(YYYYNNN CYCLESTARTDATE(87/04/07))
DEFINE DUMPCYCLE(NNNNNNY CYCLESTARTDATE(87/04/07))

DEFINE DUMPCLASS( WEEKLY RETPD(27) DAY(7) AUTOREUSE          -
                  DATASETRESTORE NORESET UNIT(3480)         -
                  VTOCCOPIES(4) )
```

Figure 31. DFSMShsm Commands for Automatic Incremental Backup of Primary Volumes

Notes to Figure 31 on page 70:

- The start time windows for the automatic incremental and volume dumps are the same. However, the volume dumping runs only on Mondays, and the incremental backup runs only Tuesdays through Fridays.
- Dump class WEEKLY is targeted every MONDAY. The volumes in this class are available for automatic reuse every 27 days. That means that four generations of dump copies exist for volumes dumped to this class.
- The NORESET parameter for this class is defined so that data set change bits are not reset after successful dumps of the volume. See “Specifying RESET for Full Volume Dumping” for a more detailed description of using the RESET parameter.
- The physical restore of data sets from dump copies in this class is allowed.
- Four copies of the VTOC is kept for each of the volumes dumped to this class.
- You should specify the AUTODUMP parameter when you define the storage group that contains the volumes to be dumped. You should also specify the dump class to which dumps should be directed.

Specifying RESET for Full Volume Dumping

The RESET parameter of DEFINE DUMPCLASS resets the VTOC change indicators of the data sets on the volume each time the volume is successfully dumped. However, you should use the RESET parameter with caution. We do not recommend resetting the change indicator during dumping if you are also using incremental backup. This is because a deleted data set might be protected only by a dump copy, and not be found by the RECOVER command without explicit help from the requestor.

Scheduling of DFSMSHsm-Called DFSMSdss Full Volume Dumps

DFSMSdss performs only volume level serialization during full volume dump processing. This means that data could be changing on one part of the volume while another part is being dumped. Therefore, you should specify the full volume dump cycle to run during periods when activity against the volume is at a minimum.

Figure 31 on page 70 shows that AUTOBACKUPSTART and AUTODUMPSTART are scheduled to run in the early morning, when activity is normally low.

Using DFSMSHsm for Volume Recovery

There are several ways to use DFSMSHsm for volume recovery:

- Volume recover. DFSMSHsm recovers the most recent incremental backup copies of individual data sets residing on the volume.
- Volume restore. DFSMSHsm calls DFSMSdss full volume restore to recover an entire volume.
- A combination of volume restore and incremental data set recovery. DFSMSHsm calls a DFSMSdss full volume restore and then recovers any more recent incremental backup versions of individual data sets.

To recover volumes that contain data that is backed up by DFSMSHsm, we recommend that you use DFSMSHsm invocation of DFSMSdss full volume restore, and DFSMSHsm incremental volume recovery. You are able to provide a more accurate reconstruction of the volume's contents with this combination. This method also provides:

- Incremental volume recovery immediately after a full volume restore. DFSMShsm compares the VTOC copies taken by the DFSMShsm full volume dump and incremental backup processes. If the incremental volume data set is more recent than that of the full volume dump, data sets restored by the full volume restore, but not on the volume during incremental backup, are deleted. Any incremental data set backups made after the full volume dump are restored. Incremental volume recovery does not recover a data set if a more recently created data set of the same name exists on the volume.
- A check of the catalog status of restored data sets. This check is done to synchronize conditions for migrated or deleted data sets, since the last backup of the data set.

DFSMShsm fails the restore if you use DFSMShsm to call DFSMSdss to recover a system-managed volume, and the dump copy being used is a copy of the volume when it was not managed by the system.

Specify the RECOVER command as follows to perform volume recovery with DFSMShsm:

```
RECOVER * TOVOLUME (PRM003) UNIT (3390) -
        FROMDUMP (DUMPCLASS (WEEKLY) APPLYINCREMENTAL)
```

Figure 32. Recovering a Volume with DFSMShsm Incremental Recovery

The APPLYINCREMENTAL parameter requests that an incremental volume recover process follows the full volume restore. The following steps are taken after the full volume restore if this parameter is specified:

1. Recover any integrated catalogs on the volume.
2. Recover the most recent backup copy of data sets cataloged in integrated catalogs.
3. Delete any restored data sets that have been migrated or deleted since the dump of the data sets was made.
4. Recover any incremental data set backups that were made after the full volume dump was performed.

Recovering System-Managed Volumes

When you use DFSMShsm incremental recovery to recover a system-managed volume, the method used to assign SMS classes to a data set is determined by whether DFSMShsm **replaces** or **recreates** the data set.

- The SMS classes maintained in the catalog entry for the data set are used if a data set is replaced on the volume.
- The ACS routines are called to determine the SMS classes and storage group if a data set is created on the volume. In this case, the selected volume might not be the volume specified on the RECOVER command.

For more information on data set recovery to SMS volumes, see *MVS/ESA SML: Managing Data*.

Using DFSMSdss for Backup and Volume Recovery

For storage groups and non-system-managed volumes on which you do not use DFSMSShsm incremental backup, consider complementing DFSMSShsm-called DFSMSdss full volume dumps with DFSMSdss data set backup. Filtering on the data-set-changed flag allows you to do incremental data set backups. Specifying the RESET parameter allows you to reset the flag after the data set has been successfully backed up.

Recovery of volumes backed up using this method is a two-step process:

1. Restore the volume from the latest dump copy using DFSMSShsm-called DFSMSdss full volume restore. Use the same command shown in Figure 32 on page 72, omitting the APPLYINCREMENTAL parameter.
2. Restore the incremental data set backups that were made since the volume dump by using the REPLACE parameter of the DFSMSdss RESTORE command. REPLACE specifies that if the data set already exists on the target DASD volume, it can be replaced by that from the source dump volume.

For more information on using DFSMSdss to backup and restore data sets on system-managed volumes, see *MVS/ESA SML: Managing Data*.

Using DB2 Utilities for Backup and Recovery

This section covers, in general terms, how database volumes can be handled by DFSMSShsm and DFSMSdss in conjunction with database utilities.

IMAGECOPY

Take a regular backup copy of each of the table spaces using the DB2 IMAGECOPY utility. If you take incremental copies, run the DB2 MERGECOPY utility periodically to consolidate incremental image copies with the latest complete image copy taken, and to create a new *base-level* image copy of the table space. This makes recovery easier than having to run either multiple image copies or several MERGECOPY jobs to obtain the latest backup version.

RECOVER

You use the DB2 RECOVER utility to recover data. It gets data from the backup copies, and then adds the most recent changes from the recovery log. Each change made to the data in the table space is recorded in the recovery log. Therefore, recovery of a table space is done by merging the recovery log with the most recent full image copy of the table space to be recovered. This results in the table space being recovered to its last point of change, and therefore, ensures that data integrity is achieved. You can also recover the data to some point other than the most current, such as to a specific copy or to a specific point in the log data.

For more information about recovering DB2 table spaces, see the *DB2 Administration Guide*.

Backup and Recovery Strategies

Only the owner of the data can determine the backup requirements of a specific data set or object. Thus, you should use the requirements specified by the users to develop a global backup and recovery strategy. Some data is easily recreated and should never need backup. Other data is critical to the smooth operation of the system or of your business, and should be backed up frequently. You should also consider using dual copy to provide continuous availability for critical data sets.

Recommended backup strategies for various system-managed storage groups and data types are outlined below.

Primary and Special Application Storage Groups

Use management class to provide DFSMShsm automatic backup of individual data sets. Specify AUTO DUMP on the storage group definition to use DFSMShsm invocation of DFSMSdss full volume dump to automatically dump volumes. Use DFSMSdss or DFSMShsm inline backup (PGM=ARCINBAK) to provide special batch application backup requirements (for example, at specified points during a batch update cycle). Some applications might also use generation data group processing to provide backup.

Large Storage Group

Use management class to provide automatic DFSMShsm backup of individual data sets directly to tape. Specify AUTO DUMP on the storage group definition to provide DFSMShsm invocation of DFSMSdss full volume dump to automatically dump volumes. Multivolume data sets can be backed up by DFSMShsm in place of scheduled DFSMSdss backups. You can back up multivolume VSAM data sets using DFSMShsm as the data mover.

Database Storage Group

DFSMShsm-called DFSMSdss full volume dumps can complement the database utilities specific to the product being used (for example, image copy). The database utilities normally retain the logical relationships of databases and maintain the integrity of multivolume databases. For those reasons, the database administrator might use only the database recovery utilities to recover database volumes.

System Data

Your system programmer should provide backup copies of critical system data sets to ensure easy recovery in case of a system failure. Use DFSMShsm or DFSMSdss to back up other system data sets. To back up volumes, call DFSMSdss full volume dump using DFSMShsm.

There are special considerations that apply to backup and recovery of catalogs and control data sets (such as the RACF data set). In these cases, data set backup can be complemented by techniques such as duplexing and journaling. For more information on this, see *MVS/ESA SML: Managing Data*.

Unmovable and ISAM Data

Use DFSMSdss on a scheduled basis to back up unmovable data sets. Call DFSMSdss full volume dump through DFSMShsm to dump volumes.

Managing Space in Storage Groups

One of your main tasks is to provide enough free space in storage groups for new data set allocation while ensuring that DASD space is used efficiently.

Your service-level agreement should detail the amount of space users can expect to be available on a daily basis in each storage group for new data set allocation. You should set a *target* for each storage group indicating when available space reaches the minimum level specified in the service-level agreement. The target is defined as the occupancy threshold. You can use the access method services DCOLLECT command to monitor free space at both volume and storage group level. For more information about using DCOLLECT, and SLR for free space recording, see “Monitoring Pool Storage Groups” on page 64 and *MVS/ESA SML: Managing Data*.

Use management class to reclaim wasted space, delete unwanted data sets, migrate data sets that are not referenced, compress partitioned data sets, and consolidate free space on volumes. This provides the required amount of free space, and ensures efficient use of DASD in storage groups. The sections that follow describe this technique.

Release allocated but unused space: A lot of space is typically wasted by over-allocation. A user who allocates 40 tracks for a data set on a DASD volume and uses only 12 tracks, leaves 28 empty tracks; only 30% of the space allocated to the data set contains valid data. Those 28 tracks of unused space not released for use by other jobs are wasted.

Delete obsolete and unwanted data sets: Unwanted temporary or expired data sets should be deleted from storage groups.

Migrate data sets that are not referenced: Data sets that are not referenced data sets should be migrated from storage group volumes and stored in compacted form on migration volumes.

Compress partitioned data sets to remove imbedded unused space: Deleting or updating members within a partitioned data set results in unused space within the extents of the data set. Compressing the data set removes unused space within partitioned data sets.

A better way to manage space is to convert existing partitioned data sets to PDSEs. A PDSE does not require compression. The space becomes available for allocation of a new member when a member is moved or deleted from a PDSE, without having to compress or reorganize the entire data set to consolidate the fragmented space for reuse. For more information about PDSEs, see *MVS/ESA SML: Managing Data* and *DFSMS/MVS V1R2 Using Data Sets*.

Defragment volumes to consolidate free space: Space is also wasted from fragmentation. Free space on the volume is fragmented into small pieces as data sets are created, deleted, moved, and extended. New data set allocations could fail because there is not enough contiguous free space to allocate a data set in five or fewer extents.

Clear spill volumes: Develop a regular schedule for returning data sets from the spill pool to volumes in the storage group where the data sets should reside. One way to transfer them is to use the DFSMSdss COPY utility when space becomes available.

Managing Space with DFSMShsm

When DFSMShsm performs primary space management against system-managed volumes, it attempts to create enough free space so that the total amount of allocated space on the volume equals the low occupancy threshold specified in the storage group definition. DFSMShsm selects storage groups for automatic processing if:

- AUTO MIGRATE is specified in the storage group definition
- The SMS status of the volume is not QUIALL or DISALL
- The MVS status of the volume is ONLINE

The system-managed volumes to be processed are selected from the storage groups associated with the SMS configuration in which DFSMSHsm is running.

Space management requirements for a data set are specified in the management class for the data set. DFSMSHsm can manage space automatically by deleting temporary and expired data sets, releasing allocated but unused space, compressing partitioned data sets, and by migrating data sets. For information about using the management class to specify DFSMSHsm processing requirements and how DFSMSHsm processes system-managed volumes, see *MVS/ESA SML: Managing Data*.

You can reclaim wasted space in a storage group through automatic DFSMSHsm space management. This ensures enough free space for new allocations. The following parameters, set in the DFSMSHsm member of SYS1.PARMLIB, can help you use space more efficiently in storage groups and migration volumes:

MAXEXTENTS

A SETSYS parameter that specifies the number of extents that, if equaled or exceeded, causes DFSMSHsm to reallocate all non-VSAM data sets whenever volume migration occurs. Extent reduction (migration with immediate recall) releases unused space, reduces the number of extents, and compresses partitioned data sets.

Normal recall of a non-VSAM data set attempts to allocate a primary allocation size equal to the space used by the original data set. You could consider this *extent reduction*, because the whole data set is now allocated in one extent, but it is not governed by the MAXEXTENTS parameter.

COMPACT

Requires that data be compacted by software before it is stored on migration or backup volumes. DFSMSHsm does not perform compaction on sequential and key-sequenced data sets that are already compressed.

Compacting data can reduce the amount of storage required for migration and backup by as much as 50%, even after wasted space is removed. You should use software compaction for DASD volumes in migration levels 1 and 2. However, because compaction involves some processor overhead, it is more efficient to use hardware compaction when moving data to tape.

SMALLDATASETPACKING

Allows DFSMSHsm to store very small physical sequential data sets as records in a VSAM key-sequenced data set on migration level 1 volumes. For more information, see *DFSMS/MVS V1R2 DFSMSHsm Storage Administration Guide* and *DFSMS/MVS V1R2 DFSMSHsm Implementation and Customization Guide*.

OPTIMUMDASDBLOCKING

Allows DFSMSHsm to store the maximum data on each track of DFSMSHsm-owned DASD volumes. DFSMSHsm uses a block size that is determined by the device type for each of the owned DASD devices when you specify this option.

CONVERSION (REBLOCKTOANY)

Requires that data sets be reblocked during recall and recovery by DFSMSHsm. Reblocking data sets enables you to make the most efficient use of available space. DFSMSHsm does not request DFSMSDss to reblock a compressed sequential or key sequenced data set when performing recall and recovery.

DFSMSHsm Processing for Managing Space

Initially, DFSMSHsm attempts to free space without having to migrate data sets by deleting unwanted data sets and releasing unused space from data sets. If needed, DFSMSHsm then migrates data sets. During space management, DFSMSHsm processing:

1. Deletes data sets that have expired.
2. Releases allocated but unused space for physical sequential and partitioned data sets that have this management class specification.
3. Migrates and recalls data sets that are candidates for extent reduction (as specified by the SETSYS MAXEXTENTS command).

If the low occupancy threshold has not been met, DFSMSHsm performs data set migration. DFSMSHsm uses the data set management class to determine candidates for migration. It first migrates generation data sets that are marked as candidates for early migration. This is determined by the generation data group attributes in the data set's management class.

The migration priority of remaining candidate data sets is determined by an algorithm that considers the size and age of the data sets. The age that is used is the difference between the number of days the data set has been unreferenced and the value specified in the management class for the number of days not referenced on primary volumes. This calculated age is used because different data sets on the same volume can have different management classes, and the data sets can become eligible for migration at different times, regardless of their inactive age.

For example, if data set A goes unreferenced for 10 days, and the management class allows for the data set to go unreferenced for 7 days, its calculated age is 3 days. If data set B goes unreferenced for 31 days, and the management class allows for the data set to go unreferenced for 30 days, its calculated age is 1 day. Thus, all things being equal, DFSMSHsm migrates A before B, because A's calculated age is older.

DFSMSHsm ends space management processing of the volume when the low occupancy threshold is reached, or when there are no more data sets that can be migrated.

Moving DFSMSHsm-Owned Data to New DASD

The FREEVOL command of DFSMSHsm lets you manage DASD migration volumes. One such function is to empty a DASD migration volume that is being replaced. For more information on the FREEVOL command, see *DFSMS/MVS V1R2 DFSMSHsm Storage Administration Guide*.

Many of the space management functions of DFSMSHsm can be initiated by command. However, automatic DFSMSHsm processing should take care of most of your storage management requirements.

Using DFSMSHsm Recycle Processing

You use the DFSMSHsm RECYCLE command to consolidate valid data on DFSMSHsm-owned migration and backup tapes. The amount of valid data on backup volumes decreases as DFSMSHsm deletes old backup versions of data sets. Recycle processing consolidates the valid data by moving it onto fewer tape volumes. This frees tapes for reuse and helps to reduce tape mount activity during automatic backup. Recycle processing can also be used for migration level 2 volumes.

Managing Space with ISMF

You can use ISMF to perform many DFSMSdss and DFSMSHsm storage management tasks interactively. For example, you can use ISMF to compress partitioned data sets, release unused space, or delete obsolete data sets by calling the appropriate product to perform the function. However, manual space management with ISMF should be used only in special cases to supplement the automated space management processing of system-managed data sets and volumes.

Compressing and Condensing Partitioned Data Sets with ISMF

You can use the COMPRESS or CONDENSE functions of ISMF to remove imbedded free space from partitioned data sets. The COMPRESS command or line operator uses DFSMSdss to consolidate the members of partitioned data sets; the CONDENSE line operator uses DFSMSHsm to migrate and immediately recall a data set to compress it and also release the unused space. COMPRESS runs as a DFSMSdss background job; CONDENSE runs in the foreground under DFSMSHsm control.

The COMPRESS and CONDENSE functions of ISMF are particularly useful for large partitioned data sets that are not managed by DFSMSHsm space management processing.

PDSEs do not require compressing. For more information, see “Compress partitioned data sets to remove imbedded unused space” on page 75 and *MVS/ESA SML: Managing Data*.

For more information on how to use ISMF to compress partitioned data sets, see *DFSMS/MVS V1R2 Using ISMF*.

Releasing Unused Space with ISMF

You can use the RELEASE function of ISMF to release allocated but unused space from system-managed physical sequential, extended sequential, partitioned, and VSAM extended format key sequenced data sets. The RELEASE command or line operator uses DFSMSdss to release unused space from the end of one or more data sets.

RELEASE should be used after COMPRESS to release all unused space for partitioned data sets. ISMF RELEASE runs as a DFSMSdss background job, which can be useful for data sets that do not specify DFSMSHsm space management processing in their management class definitions.

For more information on how to use ISMF to release unused space, see *DFSMS/MVS V1R2 Using ISMF*.

Consolidating Space on Fragmented Volumes with DFSMSdss

Free space on a volume can become heavily fragmented as data sets are created, deleted, moved, and updated. Fragmentation results in inefficient use of DASD space, and as a result, causes space-related abends and degraded performance. Abends occur because, although there might be enough total free space on the volume for allocation, there might not be enough contiguous free space to satisfy a particular allocation or extent request. Performance suffers from excessive DASD arm movement when free space is scattered rather than consolidated.

Space fragmentation on a volume is measured by a *fragmentation index*, a three-digit decimal value between 0 and 1. The higher the value, the more severe the space fragmentation. Although the index certainly varies by the level of activity on the volume, a value above 0.5 generally indicates excessive space fragmentation.

You can use the DEFRAG command of DFSMSdss to check the fragmentation index for a given volume, and to consolidate fragmented space if the index reaches a certain level. DEFRAG reduces space fragmentation by relocating data set extents to assemble several groups of contiguous free space.

You can use the DFSMSHsm space management volume exit (ARCMVEXT) routine to check the fragmentation index on a volume during DFSMSHsm automatic space management processing for storage groups that specify DFSMSHsm automatic space management. Also use this routine to submit a DFSMSdss DEFRAG job if the index is too large. For more information on the DFSMSHsm space management volume exit routine and an example of the exit used to schedule DEFRAG, see *DFSMS/MVS V1R2 DFSMSHsm Implementation and Customization Guide*.

You can trigger a weekly DFSMSdss DEFRAG job using NetView for storage groups or non-system-managed volumes that are not processed by DFSMSHsm space management. You can also use ISMF to list the fragmentation index of a volume and submit a DEFRAG job if appropriate.

You should run DEFRAG against the DFSMSHsm migration level 1 DASD volumes periodically to ensure sufficient contiguous space for migration of large data sets. For example, you could use ISMF on a monthly basis to check the fragmentation index on the DFSMSHsm migration level 1 volumes and run a DEFRAG job if the index is greater than or equal to 0.450.

For more information on using DEFRAG to consolidate free space on a volume, see *DFSMS/MVS V1R2 DFSMSdss Storage Administration Guide*.

Using Database Utilities to Manage Space

Space management for database data can be handled by some DFSMSdss functions. However, there are also some storage management tasks that the database administrator should perform using the appropriate database utilities.

Space usage in a DB2 environment can be monitored using the STOSPACE utility program to record statistics for all volumes in a DB2 storage group. This can be equated to the database SMS storage group. The output of the STOSPACE utility helps the database administrator determine when to run REORG to reorganize the database and recover unused space. For more information on these DB2 utilities, see *DB2 Administration Guide*.

Chapter 5. Planning and Implementing Tape Storage Groups

Tapes are an effective media for storing many types of data, especially backup and archive copies of DASD data sets. However, tapes must be mounted to be used, and the capacity of tapes is often not fully used. You can use DFSMS/MVS to help you make more effective use of your tape resources.

With the Storage Management Subsystem (SMS), you can define a group of DASD volumes to act as a buffer for tape drives, and allow DFSMSHsm to manage writing the tape volumes and data sets on those volumes. With DFSMS/MVS, you can have system-managed tape volumes, and you can use the DFSMSrmm functional component of DFSMS/MVS to manage the inventory of system-managed and non-system-managed tapes.

Managing Tape in the DFSMS Environment

DFSMS/MVS offers you a number of tools to help you manage your tape volumes. To use these tools effectively, we recommend the following approach for bringing tape volumes into system-managed storage:

1. Use a technique called *tape mount management* to help reduce your tape mounts and efficiently fill the storage capacity of your existing tapes. This includes using the Volume Mount Analyzer to analyze your tape mounts, and the improved data recording capability feature of 3480, 3490, and 3490E tape devices to increase the amount of data on each cartridge. This technique is explained in "Managing Tape Mounts" on page 82.
2. Define tape libraries so that SMS can manage tape volumes contained in the libraries. Tape libraries are explained in "Using Tape Libraries" on page 87.
3. Use DFSMSrmm to maintain tape data set information, to manage tape volumes, to manage shelf storage and offsite storage for tape volumes, and to manage the movement of tape volumes between tape libraries. For more information on DFSMSrmm, see "Managing Your Tape Inventory" on page 98.

Although we recommend implementing these steps in the order indicated, you can implement them in any order.

Managing Tape Mounts

A tape cartridge can contain several gigabytes of data. However, many tape cartridges, through inefficiency, end up containing only a few megabytes of data, leaving most of the tape cartridge empty. By using a technique called tape mount management, you can increase the quantity of data on each tape cartridge and decrease the number of tape mounts.

Understanding Tape Mount Management

Many of the data sets that reside on tape are user backups or archive copies, and are never updated or read after they are first written. A user might write no more than one or two data sets on a tape volume, leaving most of the tape volume empty. These kinds of data set are most effectively handled by DFSMSHsm.

Tape mount management takes advantage of the abilities of DFSMSHsm to migrate data from a pool storage group directly to tape, and to fill each tape cartridge with data before mounting a new cartridge. This is done by defining a buffer to hold new tape data set allocations on DASD before the data sets are written to tape.

You design your storage group ACS routine to redirect selected tape allocations to the tape mount management buffer storage group. You also assign this storage group the automatic interval migration attribute (I for AUTO MIGRATE), so that DFSMSHsm migrates data from the DASD volumes every hour if the migration threshold has been exceeded, even if the low threshold was not reached during the previous interval migration. This ensures that space is available in the tape buffer for new allocations.

You should also define a tape mount management spill buffer so that allocations of especially large data sets do not fail because of lack of space. If the high threshold of the tape mount management buffer volumes is exceeded before DFSMSHsm does interval migration, allocations can be satisfied on volumes in the spill buffer.

During interval migration, DFSMSHsm manages the tape mounting and writing so that each tape cartridge is filled to capacity.

You should also modify your storage group ACS routine to ensure that recalled data sets are not recalled to the tape mount management buffers. Ensure that recalled data sets are directed to volumes in a primary storage group. See "Writing the Storage Group ACS Routine for Tape Buffers" on page 85 for an example of redirecting recalled data sets to a primary storage group.

Not all data sets are good candidates for tape mount management. For example, data sets that need to be put on tape so they can be sent offsite should be written directly to tape.

Analyzing Tape Mounts

You can use the DFSMS/MVS volume mount analyzer to analyze your tape mounts and to help you define filtering criteria you can use in your ACS routines to redirect tape data set allocations.

The volume mount analyzer analyzes tape mount and usage information recorded in system management facilities (SMF) records, and classifies tape data according to type: for example, backup data, temporary data, or active data. Thus, the volume mount analyzer helps you identify data sets that can be routed to tape through the tape mount management buffers.

For information on using the volume mount analyzer to analyze tape mounts and to help you develop filtering criteria, see *MVS/ESA SML: Implementing System-Managed Storage* and *DFSMS/MVS V1R2 Using the Volume Mount Analyzer*.

Defining Storage Groups for Tape Buffers

There are two storage group categories used as tape mount management buffers: the tape mount management buffer and the tape mount management spill buffer.

Tape Mount Management Buffer

Pool storage groups used as a buffer to store data sets before they are moved to migration level 2 tape or migration level 1 DASD volumes. The management class assigned to the data sets determines whether data sets are moved to migration level 1 DASD volumes from the buffer, or whether they are written directly to migration level 2 tape.

Set the high threshold so that migration occurs only after enough data is on the volumes to fill at least one tape. We recommend setting the high threshold to 95%. This ensures that enough data is on the volumes to fill tapes and that migration to tape does not occur too frequently for your operations staff. Set the low threshold to 0% or a low value so that when migration occurs, most data can be removed from the storage group.

Set the AUTO MIGRATE attribute to **I** for interval migration. This allows DFSMSHsm to perform automatic interval migration every hour, even if the low threshold was not reached during the previous interval migration.

Set the SMS volume status to **ENABLE** for all volumes in the buffer. However, if you do not want to define a spill buffer, you can set the SMS volume status to **QUIALL** or **QUINEW** for some of the volumes in the tape mount management buffer. Any volumes in **QUIALL** or **QUINEW** act as spill volumes, and give you a result similar to having a separate spill buffer.

Tape Mount Management Spill Buffer

Pool storage groups used as spill storage for the tape mount management buffer. These storage groups help ensure that large data set allocations can be satisfied.

If you are using JES2, set the SMS storage group status to **QUIALL** or **QUINEW**, and the SMS volume status to **ENABLE** for volumes in the storage group. This ensures that the storage group is only used if the high threshold is reached in the tape mount management buffer. If you are using JES3, the SMS storage group status has to be **ENABLE**, so set the SMS volume status for volumes in the storage group to **QUIALL** or **QUINEW**. You can also use the SMS volume status on a JES2 system, but using the SMS storage group status is simpler.

We recommend setting the high threshold level at 75%, so that the spill buffer is more likely to accommodate large data sets. Set the low threshold at 0%, so that DFSMSHsm can keep the volumes empty.

As with the tape mount management buffer storage groups, set the AUTO MIGRATE attribute to **I** to allow for DFSMSHsm automatic interval migration.

Storage Group Naming Conventions for Tape Buffers

As with the pool storage groups used with primary DASD storage, develop a consistent naming convention for the storage groups used with tape mount management. The considerations for setting volume thresholds and not mixing device geometries within a storage group also apply to the tape mount management buffer storage groups.

We recommend you use the following naming conventions for pool storage groups used with tape mount management:

Storage Group Category	Storage Group Name
Tape mount management buffer	TMMBUFxx
Tape mount management spill buffer	TMMBFSxx

Note: Replace the x string with something you find meaningful. For example, TMMBUF90 could be the name of a storage group containing 3390 DASD volumes, TMMBUF80 one containing 3380 DASD volumes.

DASD Volume Naming Conventions for Tape Buffers

As with DASD volumes used for primary storage, develop a naming convention for the volume serial numbers for DASD volumes defined to the tape mount management buffer storage groups. We recommend using the prefix **TMM** for volumes defined to either the tape mount management buffer or spill buffer storage group. For example, volumes would be named TMM001, TMM002, and so on.

Alternately, you can distinguish between the volumes in the buffer and those in the spill buffer. For example, you could use **TBF** for volumes in the tape DASD buffer and **TBS** for volumes in the spill tape DASD buffer.

Sample Pool Storage Groups for Tape Buffers

Following is a description of the storage groups used in the sample storage group routine described in “Writing the Storage Group ACS Routine for Tape Buffers” on page 85. Figure 33 on page 85 shows the attributes we recommend for these storage groups.

TMMBUF80, TMMBUF90

The tape mount management buffer storage groups, assigned to new tape data sets that are to be allocated on DASD volumes before DFSMSHsm moves them to tape. The low threshold is set to 0% so that all data sets can be moved to tape. The AUTO MIGRATE attribute is set to **I** so that DFSMSHsm can migrate data hourly as needed.

TMMBFS80, TMMBFS90

The tape mount management spill buffer storage groups, assigned to new tape data sets that are to be allocated on DASD volumes before DFSMSHsm moves them to tape. The spill storage groups are used only when there is not enough space remaining in TMMBUF80 and TMMBUF90 to allocate the new data set. The low threshold is set to 0% so that all data sets can be moved to tape. The

AUTO MIGRATE attribute is set to I so that DFSMSshm can migrate data hourly as needed.

Figure 33. Sample Storage Groups for Tape Mount Management

Attributes	Tape Mount Management Storage Groups			
NAME	TMMBUF80	TMMBUF90	TMMBFS80	TMMBFS90
TYPE	POOL	POOL	POOL	POOL
AUTO MIGRATE	INTERVAL	INTERVAL	INTERVAL	INTERVAL
AUTO BACKUP	YES	YES	YES	YES
AUTO DUMP	YES	YES	YES	YES
DUMP CLASS	ONSITE, OFFSITE	ONSITE, OFFSITE	ONSITE, OFFSITE	ONSITE, OFFSITE
HIGH THRESHOLD	95	95	75	75
LOW THRESHOLD	0	0	0	0
GUARANTEED BACKUP FRE- QUENCY	NOLIMIT	NOLIMIT	NOLIMIT	NOLIMIT
SMS VOLUME or STORAGE GROUP STATUS ¹	ENABLE	ENABLE	QUIALL or QUINEW	QUIALL or QUINEW

Note: ¹ For JES2 systems, it is simpler to use the SMS storage group status instead of the SMS volume status for the spill buffer. For JES3 systems, you must use the SMS volume status.

Writing the Storage Group ACS Routine for Tape Buffers

Figure 34 on page 86 shows an example of a storage group ACS routine fragment that assigns tape data sets to tape mount management storage groups according to management class. The management class routine must ensure that recalled or recovered data sets are not assigned management classes beginning with the letters TMM. The management classes beginning with TMM are only assigned to data sets that are assigned to the tape mount management storage groups.

It is important that you do not allow recalled or recovered data sets to be assigned to the tape DASD buffer storage groups. Otherwise, the tape mount management buffer becomes filled with active data, or active data is quickly migrated back to tape. This defeats the purpose of having a buffer.

In this example, data sets assigned the TMMTEMP data class, indicating they are allocations of temporary tape data sets on tape, are assigned to the large storage group rather than the tape mount management buffers. Because these data sets are temporary, they are deleted at the end of the job and never need to be written to tape. The TMMTEMP data class allows for multivolume allocation.

The simplicity of this storage group routine depends on the filtering logic in the data class and management class routines. You can develop the filters used in these routines with the volume mount analyzer. *MVS/ESA SML: Implementing System-Managed Storage* explains how to use the volume mount analyzer and write the data and management class routines for tape data sets.

General-use programming interface

```
/* Start of Mainline SELECT */
/*****
SELECT
  WHEN (&DATACLAS = 'TMMTEMP')          /* Temporary tape data sets*/
                                          /* go to the LARGE group */
    DO
      SET &STORGRP = 'LARGE90','LARGE80'
      EXIT
    END
  WHEN (&MGMTCLAS = TMM*)              /* Other tape data sets */
                                          /* selected for TMM go */
                                          /* to TMM buffer; BFS */
                                          /* is a "spill" area only */
                                          /* used if BUF is full */
    DO
      SET &STORGRP = 'TMMBUF90','TMMBUF80','TMMBFS90','TMMBFS80'
      EXIT
    END
END
/* End of Mainline SELECT */
*****/
```

Figure 34. Sample Storage Group ACS Routine for Tape Mount Management. For an example of a complete storage group ACS routine that implements tape mount management with management of primary storage, see MVS/ESA SML: Implementing System-Managed Storage.

End of General-use programming interface

Using the Improved Data Recording Capability with DFSMSHsm

You can use the improved data recording capability (IDRC) feature of the 3480, 3490, and 3490E Magnetic Tape Subsystems to compact data written to tape and so increase the storage capacity of your tape cartridges.

To take advantage of IDRC for DFSMSHsm migration and backup tapes, specify the following parameters on the DFSMSHsm SETSYS command:

```
SETSYS TAPEMIGRATION(ML2TAPE(TAPE(3490)))-
BACKUP(TAPE(3490))
```

The TAPEMIGRATION and BACKUP parameters specify that 3490E tape units are to be used for migration and backup volumes, respectively. The TAPEHARDWARECOMPACT parameter is not necessary for the 3490E.

If you are using IDRC with the 3480, specify the following parameters on the DFSMSHsm SETSYS command:

```
SETSYS TAPEHARDWARECOMPACT-
TAPEMIGRATION(ML2TAPE(TAPE(3480X)))-
BACKUP(TAPE(3480X))
```

The TAPEHARDWARECOMPACT parameter enables you to use IDRC when 3480 units capable of IDRC are used. The TAPEMIGRATION and BACKUP PARAMETERS specify that 3480 tape units are to be used for migration and backup volumes, respectively.

Using Tape Libraries

A *tape library* is a collection of tape devices and tape volumes that can be used on those devices. You can associate a tape library with one or more storage groups, and a storage group with one or more tape libraries.

There are two types of tape library: automated and manual. An automated 3495 can contain a mix of 3490 and 3490E tape devices. The 3494 automated tape library contains only 3490Es. Automatic tape library dataservers allow automatic tape mounting and eliminate the need for most tape operator intervention. A *manual* tape library contains 3490 or 3490E tape devices in an IBM 3495 M10 Tape Library Dataserver, and requires operator interaction. The volumes in both tape library dataservers are considered system-managed.

DFSMSrmm can help you automatically manage the shelf storage for both automated and manual tape libraries.

To use tape libraries, you must define a tape configuration database and at least one tape storage group. You should also start the object access method (OAM) address space. For a complete discussion of OAM's role in the tape library support and installation requirements, refer to *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Advantages of Using Tape Libraries

As your storage needs grow, you might need more than one tape center available to a single system. These tape centers can be very far away from one another. Without tape libraries, the system might select a tape device at one location for a requested mount of a tape volume residing at another, distant, location.

By defining tape libraries, you can establish a relationship between the volumes kept at a location and the tape devices at that location. With tape libraries, the system ensures that requests for tape mounts are assigned to devices in the same library as the tape volume. DFSMS/MVS also ensures that cartridges are mounted on an appropriate device, based on the type of tape cartridge and the recording format.

Tape libraries automated by the 3494 or 3495 Tape Library Dataserver have an additional advantage. When a user requests a mount for a volume kept in a 3494 or 3495 Tape Library Dataserver, the dataserver mounts the volume without operator intervention. This frees your operations staff to do other tasks.

Also, a manual tape library provides SMS-tape management in an environment where the cost of an automated tape library might not be justified.

You can share both automated and manual tape libraries among systems using the Storage Management Subsystem.

Defining the Tape Configuration Database

The *tape configuration database* is one or more volume catalogs. A *volume catalog* is the basic catalog structure (BCS) of an integrated catalog that contains entries for tape volumes or tape libraries. Volume catalogs are not connected to VSAM volume data sets (VVDSSs), and they do not contain entries for individual data sets. Because volume catalogs do not contain entries for data sets, you cannot define aliases for volume catalogs.

You define volume catalogs with the access method services DEFINE USERCATALOG VOLCATALOG command. Volume catalogs must be named as follows:

hlq.VOLCAT.VGENERAL

Is the base volume catalog for the tape configuration database, and contains entries for all tape libraries you define. It also contains any tape volume entries that are not contained in the specific volume catalogs. You can use any valid name for the HLQ, and identify the qualifier you use in the LOADxx member of SYS1.PARMLIB. SYS1 is the default HLQ.

hlq.VOLCAT.Vx

Are the specific volume catalogs for the tape configuration database. These catalogs contain the entries for tape volumes whose volume serial numbers begin with the character *x*. The *x* can be any valid character for volume serial numbers, and the *hlq* must be the same as used for the general volume catalog.

You should only define specific volume catalogs if you have a large number of tapes with volume serial numbers with the same first character. When deciding on the number of specific volume catalogs, consider performance requirements for catalog access and availability requirements in case a catalog needs to be recovered. Smaller catalogs tend to have better performance, and can be recovered more quickly, than very large catalogs.

If you are going to share tape devices among more than one system, define the volume catalogs with share options (3 4) and place the catalogs on DASD volumes shared by all the systems.

Figure 35 on page 89 shows an example of defining a general volume catalog. For information on defining catalogs, and on performance and maintenance considerations for catalogs, and other system-managed storage considerations, see *DFSMS/MVS V1R2 Managing Catalogs*.

After defining the volume catalogs, use IMPORT CONNECT to connect them to the master catalogs of each sharing system.

If a volume catalog becomes corrupted, or it becomes inconsistent with the inventory kept in the 3494 or 3495, you can use the access method services CREATE and ALTER commands to add missing entries or correct existing entries.

```

//DEFINE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
    DEFINE USERCATALOG -
        ( NAME(SYS1.VOLCAT.VGENERAL) -
          MEGABYTES(10 10) -
          VOLUME(SYS303) -
          VOLCATALOG -
          FREESPACE(10 10) -
          STRNO(3) -
          IMBED -
          REPLICATE ) -
        DATA ( CONTROLINTERVALSIZE(4096) -
              BUFND(4) ) -
        INDEX ( BUFNI(4) )
/*

```

Figure 35. Defining a General Volume Catalog

Defining Tape Libraries

After you have defined the tape configuration database, you can define tape libraries with the ISMF Library Management Application. Select DEFINE from the tape library application selection menu, and fill in the appropriate values on the tape library define panel.

Figure 36 on page 90 shows an example of defining a tape library named TLB3495M. This library contains both cartridge system tapes (Media 1) and enhanced cartridge system tapes (Media 2). Scratch volume thresholds are set at 3000 and 200, respectively. By default, volumes added to the library are added as scratch. The sections that follow the example explain the general considerations for defining tape libraries.

```

                                TAPE LIBRARY DEFINE                                Page 1 of 2
COMMAND ==>

SCDS NAME:   STGADMIN.SMS.SCDS1
LIBRARY NAME: TLB3495M

TO DEFINE LIBRARY, SPECIFY:

DESCRIPTION ==> Tape library containing 3495 Tape Library Dataserver
               ==> and 3490 and 3490E drives, mixture of tape media.

LIBRARY ID           ==> 0001C   (00000 to FFFFF)
CONSOLE NAME        ==> T3495CON
ENTRY DEFAULT DATA CLASS ==>
ENTRY DEFAULT USE ATTRIBUTE ==> SCRATCH (PRIVATE or SCRATCH)
EJECT DEFAULT       ==> KEEP    (PURGE or KEEP)

MEDIA TYPE:         SCRATCH THRESHOLD

MEDIA 1:            ==> 3000    (0 to 999999)
MEDIA 2:            ==> 200    (0 to 999999)
USE ENTER TO PERFORM VERIFICATION; USE DOWN COMMAND TO VIEW NEXT PANEL;
USE HELP COMMAND FOR HELP; USE END COMMAND TO SAVE AND EXIT;
CANCEL TO EXIT.

```

```

                                TAPE LIBRARY DEFINE                                Page 2 of 2
COMMAND ==>

SCDS NAME:   STGADMIN.SMS.SCDS1
LIBRARY NAME: TLB3495M

INITIAL ONLINE STATUS (YES, NO, or blank):

SYSTEM01 ==> YES  SYSTEM02 ==> YES  SYSTEM03 ==> YES  SYSTEM04 ==>
SYSTEM05 ==>     SYSTEM06 ==>     SYSTEM07 ==>     SYSTEM05 ==>

WARNING:
When you connect a tape library to a system group rather than a system ,
you lose the ability to vary that library online or offline to the
individual systems in the system group. It is strongly recommended that
the tape library be connected to individual systems only.

USE ENTER TO PERFORM VERIFICATION; USE UP COMMAND TO VIEW PREVIOUS PANEL;
USE HELP COMMAND FOR HELP; USE END COMMAND TO SAVE AND EXIT; CANCEL TO EXIT.

```

Figure 36. Defining a Tape Library

Tape Library Naming Conventions

Develop a naming convention to help simplify tape library management. For example, if you have tape libraries in several buildings, you might include the building and room number in the library name: B86R108.

Alternatively, you might use the name to indicate the type of tape devices in the library, if you are not mixing tape devices in your libraries.

However, to avoid confusion, you should not use the following names for tape libraries, because DFSMSrmm uses these names to indicate the location of tape volumes:

DISTANT
HOME
LOCAL
REMOTE

Defining the Hardware Attributes

When you define a tape library, you must specify the library ID and optionally the console name for the library. The library ID for the Tape Library Dataserver is set at installation by the IBM customer engineer.

In addition, you can associate an MVS console with each tape library. Messages related to a specific tape library can then be directed to an MVS console near that tape library. If you do not specify a console name, messages continue being routed using the MVS routing codes. For more information, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Defining the Tape Volume Attributes

When you define a tape library, you can define some general requirements for the tape volumes contained in the library. However, it is a better practice to define volumes and their attributes to DFSMSrmm prior to entering them into a tape library. DFSMSrmm ensures that the volumes retain the appropriate attributes.

The following are the attributes related to tape volumes:

ENTRY DEFAULT DATA CLASS

specifies the data class to be used to determine the interchange attributes for private volumes entered into the tape library. No other attributes of the data class are used. You can use the cartridge entry installation exit to assign appropriate interchange values that override the default data class specifications.

ENTRY DEFAULT USE ATTRIBUTE

specifies the default *use attribute* for new volumes entered into the library. The use attribute defines whether a volume is *private*, that is, whether it contains valid data, or *scratch*, that is, whether it is available for a scratch mount.

EJECT DEFAULT

specifies the default volume record disposition for volumes ejected from the library. You can specify that the volume record be kept, even though the volume is no longer in the library, or you can specify that the volume record be purged. For example, you might want to keep records for ejected cartridges if you are ejecting them from a library to put them in shelf storage. The information for ejected tapes specifies that the tape resides on the shelf, rather than in a specific library.

SCRATCH THRESHOLD

specifies the number of scratch volumes that should be maintained in the library. Once the number of scratch volumes in the library falls below the scratch threshold, a message is issued to the operator indicating that more scratch volumes should be entered into the library. The message is deleted once the number of scratch volumes in the library exceeds twice the scratch volume threshold. Set the thresholds to ensure that enough scratch tapes are available for normal tape processing.

You can specify separate thresholds for cartridge system tapes (Media 1) and enhanced cartridge system tapes (Media 2). This allows you to mix cartridge capacities in a tape library. Only specify a scratch threshold for the types of media you are using in the library.

As you enter tapes into a library, or eject tapes from a library, the tape configuration database is updated. It is also updated as scratch volumes are used and changed to private volumes, and as private volumes are returned to scratch.

Defining Tape Storage Groups

For DFSMS/MVS to manage a tape volume, the volume must belong to a storage group. Figure 37 shows an example of defining a tape storage group.

```

                                TAPE STORAGE GROUP DEFINE
COMMAND ==>

SCDS NAME:          STGADMIN.SMS.SCDS1
STORAGE GROUP NAME: TAPEHSM

TO DEFINE STORAGE GROUP, SPECIFY:

  DESCRIPTION ==> Storage group for DFSMShsm migration level 2 and
                ==> backup tape volumes

  LIBRARY NAMES (1 to 8 characters each):
    ==> TLB3495M ==>          ==>          ==>
    ==>          ==>          ==>          ==>

  DEFINE SMS STORAGE GROUP STATUS   ==> N (Y or N)

USE ENTER TO PERFORM VERIFICATION AND SELECTION;
USE HELP COMMAND FOR HELP; USE END COMMAND TO SAVE AND EXIT; CANCEL TO EXIT.
```

Figure 37. Defining a Tape Storage Group. The storage group TAPEHSM uses tape library TLB3495M.

A tape library dataserer can be used alone or in conjunction with another tape library dataserer. You can define tape storage groups that consist of:

- Automated tape library dataserers
- Manual tape library dataserers
- A combination of automated and manual tape library dataserers

Unlike dummy or pool storage groups, you cannot connect a group of volumes with a tape storage group when you define the group. For tape storage groups, you only specify the tape libraries that are to contain volumes belonging to the storage group. Up to eight tape libraries can contain volumes in a single storage group.

However, if you enter tapes containing a multivolume data set into a storage group, ensure that all volumes are entered into the same tape library. All volumes of a multivolume mount request must reside in the same tape library.

When a data set is written on a scratch tape, the tape's use attribute is changed to private, and the tape is added to the storage group assigned to the data set. You can explicitly assign private volumes to a storage group with DFSMSrmm. Scratch volumes cannot be assigned to a storage group.

You can create a private volume (a volume with data on it, not to be used as a scratch tape) by entering a null storage group when assigning a volume to a library. You do not need to write an entry installation exit or define a private volume to DFSMSrmm before you enter it into a library.

A single tape library is not limited to one storage group. More than one storage group can own volumes contained in a single tape library. Normally, however, you should use the entry installation exit and assign a storage group.

See *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference* for complete information on defining storage groups.

Determining the Number of Tape Storage Groups

Before you define your tape storage groups, decide if you need to separate tape volumes into different storage groups. As with pool storage groups, the fewer groups you define, the easier it is to maintain them.

However, there are some benefits to defining more than one tape storage group. Consider defining a tape storage group for each of the following categories, if they apply:

Offsite or Interchange

Tape volumes that are created to be moved offsite. This category might include data that is being sent to other companies, to other sites within your company, or to a disaster backup center.

DFSMSShsm Tapes

Tape volumes used by DFSMSShsm for backup and migration. Defining a separate storage group for DFSMSShsm can help you do capacity planning. You might also find it beneficial to define a high scratch level for the tape library containing the DFSMSShsm tapes, to ensure there are always enough tape volumes for daily space management.

Because DFSMSShsm tape use is often done at night, consider associating this storage group with a 3494 or 3495 Tape Library Dataserver.

General Use

Tape volumes used for general allocations that are not redirected to DASD volumes. These volumes might contain large data sets, test data sets, and other special case data sets.

Critical Applications

Tape volumes used by applications critical to your business. These applications can require special security needs, or other needs that could require separate handling.

Tape Storage Group Naming Conventions

Develop naming conventions for your tape storage groups to simplify the task of managing storage groups and writing and understanding your ACS routines.

We recommend the following naming convention for tape storage groups:

Storage Group Category	Storage Group Name
Tape	TAPExxxx

Note: Replace the xxxx string with something you find meaningful. For example, TAPEHSM could be the name of the storage group you use for DFSMSHsm migration level 2 tape volumes.

Adding Tape Volumes to a Tape Library

Once you have defined your tape libraries and tape storage groups, you need to add tape volumes to the tape libraries. If an entry for the volume exists in the tape configuration database, DFSMS/MVS verifies that the volume does not reside in another library. If there is not an entry for the volume, DFSMS/MVS adds the appropriate entry, marking the volume as private or scratch according to the default you defined for the library.

You can use DFSMSrmm to assign tape volumes to a tape library. DFSMSrmm updates its removable media inventory. When a tape volume is entered into a library and there is no entry for the volume in the tape configuration database, the system creates the volume entry, based on the information defined to DFSMSrmm. Besides assigning the volume to a library, you can assign it to a storage group if it is a private volume, and assign it the appropriate use attribute. You can also use DFSMSrmm to maintain information concerning the shelf location of the volumes. It is easier to first define a volume to DFSMSrmm so that the correct information is available to the tape library when the volume is entered into the library. DFSMSrmm allows you to add volumes to libraries in groups rather than one by one.

You can also use ISMF to change the use attribute, storage group, and other characteristics of system-managed tape volumes.

Using Tape Cartridges with Different Capacities

IBM offers two types of tape cartridge: Cartridge System Tape and Enhanced Capacity Cartridge System Tape. You can use both types of tapes within a tape library or storage group.

Use the data class to control the use of the different types of tape cartridge. The data class interchange attributes allow you to specify the type of cartridge and the recording method that is to be used for a data set. By using these attributes, you can ensure that tapes that are to be used or shipped offsite are recorded in a way that can be read using the tape devices available at the other site.

You can also use the COMPACTION attribute in the data class to specify whether the data should be compacted by the tape device when it is written.

The data class assigned to the first data set written on a tape volume determines the interchange and compaction characteristics of the volume. You cannot use different recording methods for data sets on a single volume. The entire tape volume is written using the same recording method and compaction.

Tape Volume Naming Conventions

As with DASD volumes, you should develop a naming convention for tape volume serial numbers. This makes it easier to manage tape volumes. With DFSMSrmm, the naming convention you develop for the tape volumes are also used to assign spaces for the volumes on your shelves.

Tape volumes used in a tape library are limited to using the characters A through Z and the numbers 0 through 9 in the volume serial number. However, DFSMSrmm can handle any volume serial number allowed by the MVS/ESA operating system, and you can use DFSMSrmm to initialize volumes rather than the IEHINITT utility.

Because you cannot control which scratch tape is mounted for a scratch request, you cannot easily develop naming conventions that associate the volume serial number with the content of the tape volume. Instead, develop a naming convention that associates the tape volume with the library containing the volume.

For example, you can use the letter T as the first character of the volume serial numbers for all cartridges kept in a 3495 Tape Library Dataserver. If you have a large number of volume serial numbers that begin with the same character, consider defining a specific volume catalog to contain the entries for the volumes.

No matter what convention you develop, you might have to manage tape volumes whose volume serial numbers do not match your naming convention. For example, you might receive volumes from installations with different conventions. DFSMSrmm can manage volumes even if the volume does not match your naming convention, so long as the volume serial number (volser) does not duplicate one you already have. Make sure that the volser you assign to a tape volume is not the same as an existing DASD volser.

Cataloging Tape Data Sets

You are not required to catalog tape data sets. However, we recommend that you catalog tape data sets, so that you do not need to specify a unit and volume serial number when allocating an existing data set.

Unlike system-managed data sets on DASD volumes, tape data sets are cataloged during disposition processing, rather than when they are first allocated.

DFSMSrmm maintains information about the data sets on every volume defined to DFSMSrmm, even if the data sets are not cataloged. This allows you to maintain information on data sets that reside offsite, even if you do not want to maintain catalog records for the data sets.

Writing the Tape Storage Group ACS Routine

Figure 38 on page 97 shows an example of a storage group ACS routine fragment for assigning data sets to tape storage groups. The routine assigns data to storage groups, based on data set name or the data class assigned to the data set. You can use the &LABEL variable to ensure that only new data set allocations specifying IBM standard or ISO/ANSI labels are directed to tape storage groups. For an example of a data class routine that assigns these data classes, see *MVS/ESA SML: Implementing System-Managed Storage*.

The routine first assigns DFSMSHsm migration and backup data to the TAPEHSM storage group. DFSMSHsm-written data is identified according to data set name.

Then, the routine assigns data intended for offsite storage or interchange with other installations to the TAPEOFFS storage group. Data sets created for onsite purposes are assigned to the TAPEGEN storage group.

The storage group ACS routine is only called when a data set is being written to a scratch volume. If a tape volume is already assigned to a storage group, the storage group ACS routine is not called. Once a volume becomes a private volume, users can allocate data sets on the volume if they specify the volume serial number in their mount request and they have RACF authorization to the volume. The specification for guaranteed space in the storage class is always ignored for tape allocations.

General-use programming interface

```

/*****
/* Filter list for DFSMSshm migration, backup, and dump data sets*/
*****/

FILTLIST MIGBKDMP    INCLUDE(**.HMIGTAPE.DATASET,
                           **.BACKTAPE.DATASET,
                           *.DMP.*.V*.D*.T*)

SELECT

  WHEN (&DSN = &MIGBKDMP)          /* DFSMSshm tape allocations */
  DO
    SET &STORGRP = 'TAPEHSM'
    EXIT
  END

  WHEN (&DATACLAS = 'TAPOSITE')    /* Data to be sent offsite */
  DO
    SET &STORGRP = 'TAPEOFFS'
    EXIT
  END

  WHEN (&DATACLAS = 'TAPACTV' OR &DATACLAS = 'TAPBKUP')
                                /* Active tape data and user */
                                /* backups */
  DO
    SET &STORGRP = 'TAPEGEN'
    EXIT
  END

END /* END SELECT STATEMENT */
```

Figure 38. Sample Storage Group ACS Routine Fragment for Tape Data Sets

End of General-use programming interface

Managing Your Tape Inventory

You can use the DFSMSrmm functional component of DFSMS/MVS to maintain an inventory of tape volumes and data sets. DFSMSrmm can maintain the inventory of both system-managed and non-system-managed tape volumes, and can manage volumes, based on the names of the data sets that reside on the volumes.

For system-managed tape volumes, DFSMSrmm automatically records information about the tape volumes and data that are added to tape libraries in the automated and manual tape library dataservers.

DFSMSrmm can maintain information about all your tapes, whether they are used onsite, stored offsite, or sent to other installations. When you need to find a volume, DFSMSrmm can tell you where it should be.

Following are some of the tasks you can complete with DFSMSrmm:

- Automatically assign a shelf storage location for tape volumes.
- Manage the shelf space for tape volumes, whether or not they are in tape libraries, and optical volumes that reside on the shelf.
- Control the movement of volumes between libraries and the ejection of volumes from libraries.
- Add volumes to a manual tape library in groups of volumes or one by one.
- Manage expiration and return-to-scratch for private volumes.
- Maintain information about tape volumes even if the entries for the volumes are removed from the tape configuration database.
- Manage different types of tape media, from tape reels to different types of tape cartridge.
- Create vital record specifications to manage the expiration and movement of tape volumes. Your vital record specifications override any other expiration date assigned to the volume.
- Share the tape inventory between systems, even if the systems do not share storage.
- Ensure that only valid scratch volumes are used for non-specific mount requests. This prevents users from unintentionally overwriting a volume with valid data, for example, an DFSMSHsm backup volume, if an operator mistakenly mounts a private volume.
- Create reports on your removable media. You can use DFSORT to customize the report extract file for your specific requirements.

For more information on using DFSMSrmm, and on what you can do with DFSMSrmm, see *DFSMS/MVS V1R2 DFSMSrmm Guide and Reference*.

Chapter 6. Planning and Implementing Object Storage Groups

Objects differ from data sets in that they consist of byte-stream data—they have no internal record or field structure, and, once written, can never be changed or updated. Because all objects must be system-managed, you must define object storage groups if you use them.

Managing Space with OAM

OAM provides storage management for objects. OAM manages the space automatically on DASD volumes contained in an object storage hierarchy by deleting expired objects, and, as specified by the class transition attributes, by moving objects to the next level of the hierarchy. OAM also manages space within an optical library, removing the least-recently-written volume from the library when the library becomes full. The storage management requirements for objects are specified in their management classes, and are processed by OAM during its storage management cycle.

OAM does not provide space management at a volume level. The objects themselves, however, can be identified for automatic backup and class transition. For further discussion of how to use management class to specify OAM processing requirements and how OAM processes objects in a storage hierarchy, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*, and *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

How Object Data is Stored

Objects are stored in an object storage hierarchy that can consist of DB2 table spaces stored on DASD volumes, tape and optical volumes that can either reside in libraries or on a shelf. However, the table spaces, and hence the database, can be stored over several different DASD volumes that belong to more than one DB2 storage group. Some of these DASD volumes might belong to an SMS storage group as well. Therefore, they can be managed by DFSMSHsm.

Storage groups for objects are called *object storage groups*. Object storage groups can consist of a combination of DB2 table spaces stored on DASD volumes, and up to eight optional optical libraries or pseudo libraries to handle optical disks from the shelf. Object storage groups can also be directed to standalone tape or tape libraries by using the SETOAM command in the CBROAMxx parmlib member. For more information, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*. Objects can be stored in an object storage group either individually or as a member of an object collection. One or more object collections can be in an object storage group.

It is also possible for some of the DASD volumes in one DB2 storage group to belong to one SMS storage group, while DASD volumes in another DB2 storage group could be defined to a second SMS storage group. It is possible, then, to have an overlap between an object storage group, a pool storage group, and a DB2 storage group. Figure 39 on page 100 shows the object storage hierarchy.

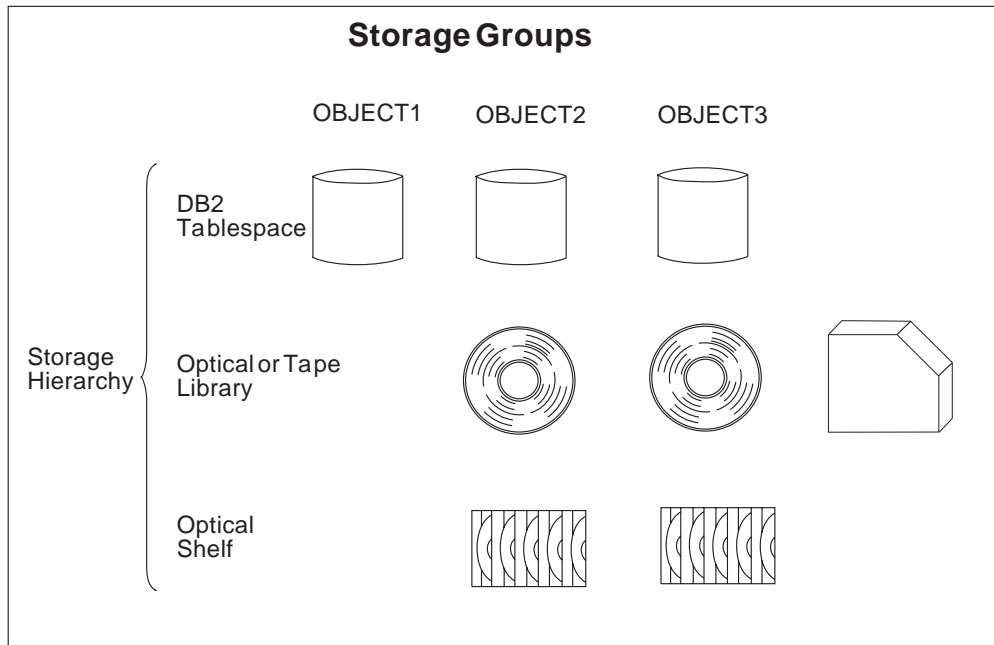


Figure 39. Object Storage Groups Stored in an Object Storage Hierarchy

Determining Storage Group Requirements for Object Data

Objects that share the same performance and availability requirements can be defined to a storage group as a member of an *object collection*. Object collections are cataloged in integrated catalogs. Object collections can only reside in a single object storage group; they cannot span groups.

Consider creating an integrated catalog for each object storage group; then catalog object collections associated with that storage group in the integrated catalog.

For more information about attributes, about defining object and object backup storage groups, and about optical libraries and drives, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Storage Group Naming Conventions and Volume Serial Numbers

Develop a naming convention to identify the object storage groups that you set up and the volumes that belong to each storage group.

As with all other tape volumes, the volumes that contain primary or backup copies of objects should follow the same naming scheme. See "Tape Volume Naming Conventions" on page 95 for more information.

Optical volumes are like tape cartridges in that you cannot control scratch mounts according to volume serial number. As with tape volumes, develop an optical volume naming convention that relates the optical volumes to the library that contains them.

For the DASD volumes containing the DB2 table spaces for objects, you might use your existing naming conventions for DB2 volumes, or develop a separate naming convention. For example, you could use OBJ as a prefix for DASD volumes used to store objects.

Object Storage Group Types and Attributes

Use the ISMF Storage Group Application to define object and object backup storage groups for objects. For a detailed explanation of how to define storage groups, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

Before you define the object or object backup storage groups, you must define your optical libraries and drives.

SMS recognizes the following types of object storage groups:

Object A storage group consisting of DB2 DASD table spaces and, optionally, one to eight optical disk libraries that are used for storing objects.

Object Backup

A storage group consisting of one to eight optical disk libraries that are used for storing backup copies of objects. One of these libraries can be a pseudo library.

There can only be one object backup storage group for each SMS system.

When you define an object or object backup storage group, you must specify the following:

- The name (QUALIFIER) of the DB2 table spaces that contain the object directory for an object storage hierarchy, and the names of the objects themselves.
- The names of the optical libraries belonging to the storage group. The libraries can be either *real* libraries defined in an optical configuration, or a *pseudo* library that represents operator accessible optical disk drives.

Note: To direct your OBJECT or OBJECT BACKUP storage group to tape, you must use the SETOAM command of the CBROAMxx parmlib member.

- The storage group status to your systems. Each storage group can only be attached to one system.
- Values for VOLUME FULL THRESHOLD, DRIVE START THRESHOLD, and VOLUME FULL AT WRITE ERROR for the optical volumes in the group.
- The CYCLE START TIME and CYCLE END TIME for OAM to start its storage management cycle.

Object and Object Backup Storage Group Connectivity

An object storage group can only be defined and connected to one system. Therefore, it is not possible to share devices in an object storage group between systems in a multisystem environment.

You can have one or more object storage groups defined to an individual system. However, only one object backup storage group can be defined for a system, and it cannot be shared between systems.

Use the ISMF Storage Group Application to define the status of a storage group to the systems that have access to it. The status indicates whether the storage group is connected to a system and the type of access that is allowed. The only system status values for object and object backup storage groups are NOTCON, ENABLE, DISALL, and DISNEW. For more information about how to specify system status values, see *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

Sample Storage Group Definitions for Objects

Figure 40 shows the sample storage group attributes and the libraries to support the target DFSMS environment for objects.

Figure 40. Sample Object and Object Backup Storage Groups

Attributes	Object and Object Backup Storage Groups			
NAME	OBJ00	OBJ01	OBJ02	BACKUP
TYPE	OBJECT	OBJECT	OBJECT	OBJECT BACKUP
QUALIFIER	GROUP00	GROUP01	GROUP02	—
CYCLE START TIME	00	13	20	—
CYCLE END TIME	03	16	23	—
LIBRARY NAMES	—	OPLIB01, OPLIB02, OPLIB03	OPLIB01, OPLIB03, OPLIB04, OPLIB05	OPBKP01, OPBKP02, OPBKP03, OPBKP04
VOLUME FULL THRESHOLD	—	32	32	04
DRIVE START THRESHOLD	—	099	999	099
VOLUME FULL AT WRITE ERROR	—	YES	YES	YES

Note: OBJ00 contains no optical libraries; therefore, it only requires parameters for the cycle start and end times.

Optical Library

An *optical library definition* is a set of attributes that you use to define an optical storage library configuration.

You must define an optical library before it can be referred to by a storage group. To establish the required definition, you must determine values for:

- Library name
- Library type
- Online status
- Operational status
- Current path
- Primary channel-to-channel address
- Primary port address
- Alternate channel-to-channel address
- Alternate port address

Optical Disk Drive

A *drive definition* is a set of attributes that you use to define an optical disk drive as a member of an optical library or as an operator accessible unit.

You must first define a library before you can define an associated drive. Defining a drive requires that you determine values for:

- Drive name
- Library name
- Drive type
- Drive number
- Online status
- Operational status
- Channel-to-channel address
- Small computer system interface (SCSI) address
- Logical unit number.

For more information on optical disk drive requirements see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Tape Library

OAM also stores object data on tape volumes if the performance matches the request. The tape volumes can reside in an automated or manual tape library data server. For information on establishing tape library definitions, see Chapter 5, “Planning and Implementing Tape Storage Groups” on page 81. For information on storing object data on tape, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Storage Class

You should identify application performance requirements when you plan your storage classes. Also consider the performance capabilities of the devices to meet them, based on those requirements.

Use storage class to determine the performance objectives of object data. The INITIAL ACCESS RESPONSE SECONDS parameter indicates the time that an application waits before object data transfer occurs. OAM uses this attribute in the storage class selection by ACS routines for objects.

Use a value of 0 if you want an object to reside on DASD, where it is stored in a DB2 table space. If you use a value between 1 and 999 for the attribute, the object resides on an optical or tape volume.

After class transition, the initial storage class assignment might change as a result of the ACS routines called by the class transition event. For more information about class transition, see “Management Class” on page 104.

The initial storage class of the object might also change if a user specifically requests, in an application program, that another storage class be assigned to the object. ACS is called to validate the storage class request and, if valid, assigns the new class.

Management Class

You use management class to identify OAM object management criteria. The management class definitions you create let you control storage and availability management at the object level.

You can also use management class to determine whether an object is to be backed up. Backup is performed by OAM, and backups are written to the object backup storage group. OAM provides backup at the object level only.

For further information about objects, see *MVS/ESA SML: Managing Data, DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*, and *DFSMS/MVS V1R2 DFSMSdfp Storage Administration Reference*.

Objects stored in an object storage hierarchy should reside on the appropriate device type (DASD, tape or optical disk) to satisfy the initial access time. This is controlled by the class transition attributes defined in management class. The attributes are used to trigger movement (or transition) of the objects by OAM to the next appropriate level of the storage hierarchy.

The OAM storage management component provides class transition attributes that work with ACS. The attributes allow you to change storage management class status, such as where the data resides, its deletion, or backup. Consider management class status requirements, based on object age, time since the last reference, or the calendar. The transition attributes can make your storage management strategies more effective.

Optical Volumes

There are three types of optical volumes: scratch, group, and backup.

Scratch

A scratch volume is an optical volume not assigned to a group. An optical volume becomes a scratch volume in one of two ways:

- An unlabeled optical volume is labeled on an operator accessible optical disk drive in response to an OAM LABEL command.
- An unlabeled optical volume is labeled on a library-resident optical disk drive as a result of being inserted into the input/output station of an optical library.

In either case, the operator is asked to supply the volume serial number for each side of the optical disk. The volumes are marked as scratch volumes in the optical configuration database.

Group

Groups of optical volumes are associated with an object storage group. Objects belonging to one group cannot share a single optical volume with objects from another group; that is, a volume must contain objects from a single object storage group.

A scratch volume becomes a group volume when it is used by the library control system to satisfy a write request that specifies an object storage group name. When a scratch volume becomes a group volume, both volumes on the optical disk become group volumes assigned to the same object storage group.

Backup

A backup optical volume is used for object backup copies. Backup volumes belong to the object backup storage group. They contain copies of objects that reside elsewhere in an object storage hierarchy.

A scratch volume becomes a backup volume when it is used to satisfy a write request for a backup volume. When a scratch volume becomes a backup volume, both volumes on the optical disk become backup volumes.

Optical Volumes in an Optical Library

You can load volumes into an optical library that have either been labeled previously on an operator accessible optical disk drive, or that are not labeled. Each volume you load is labeled as it enters the library if it is unlabeled. The operator is prompted for the volume serial numbers and the owner information during the labeling process, for each optical disk. The volume serial numbers must conform to MVS volume serial numbering conventions. The owner information is free format. Only one optical disk cartridge can be entered into the library at a time through the input/output station.

Adding or Deleting Objects

Use the ISMF Storage Group Application to add or delete optical disk libraries.

- When you add optical disk libraries to make the most of availability, or to increase capacity, be sure that your storage class definitions reflect the capabilities of your hardware.
- When you delete a DASD volume from a storage group, ensure that all the data sets and objects, stored in DB2 table spaces, are first moved off the volume. If the volume is no longer managed by the system, you need to change the CONVERTED status in the VTOC. You can do this by using the DFSMSdss CONVERTV NONSMS command. You can also reinitialize the volume by using the ICKDSF INIT command without the STGR parameter.

Defining the Optical Configuration Database

The optical configuration database contains information about the optical disk libraries in a DB2 database, optical disk drives, and optical volumes defined in the configuration. You create the database using the CBR SAMP member supplied with SYS1.SAMPLIB.

You can make optical library disk drives and libraries part of the SMS configuration. You use the ISMF Library Management application to define them and their associated storage groups, to any particular SCDS you specify. For more information, see *DFSMS/MVS V1R2 DFSMSdftp Storage Administration Reference* and *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Writing the Storage Group ACS Routine for Objects

Storage groups and classes for objects can be provided on ACS input from the following sources:

- Classes currently assigned to an object during class transition
- User request to create an object collection or create an object in a collection (OSREQ STORE)
- User request to change the classes associated with an object (OSREQ CHANGE)
- A class transition for an object.

You can use ACS to verify these specifications, and override them if you choose.

For complete information on writing ACS routines to include considerations for objects, see *DFSMS/MVS V1R2 OAM Planning, Installation, and Storage Administration Guide for Object Support*. Figure 41 on page 107 shows a segment from a storage group ACS routine that assigns an object to a storage group, based on the object name.

General-use programming interface

```

/*****/
/*          C H A N G E   H I S T O R Y          */
/*          =====                          */
/*          */
/*          DATE     RESP  DESCRIPTION OF CHANGE          */
/*          -----   ---  -----                      */
/*          90/12/03  GTB   1. FILTER LIST "OBJENV" ADDED. */
/*          */
/*          */
/*          */
/*          3. ADDITIONAL COMMENTS INCORPORATED TO IMPROVE */
/*          LEVEL OF DOCUMENTATION.                          */
/*          */
/*          4. CHANGE HISTORY INFORMATION ADDED.            */
/*          89/12/06  GTB   INITIAL ROUTINE CREATED.       */
/*          */
/*          */
/*          PURPOSE: THIS ROUTINE SEGMENT ASSIGNS A VALID STORAGE GROUP */
/*          OBJECT DATA.                                          */
/*          */
/*          */
/*          INPUT:  THE FOLLOWING ACS VARIABLES ARE REFERENCED:  */
/*          */
/*          */
/*          &ACSENVIR          &MAXSIZE          */
/*          &DSN                &STORCLAS        */
/*          &DSTYPE              */
/*          */
/*          */
/*          OUTPUT: ONE OF THE FOLLOWING STORAGE GROUP NAMES:  */
/*          */
/*          OBJ00          */
/*          OBJ01          */
/*          OBJ02          */
/*          */
/*          RETURN  ZERO IS THE ONLY RETURN CODE. ALLOCATIONS ARE NOT */
/*          CODES:  FAILED IN THIS ROUTINE.                          */
/*          */
/*****/

/*****/
/* FILTER LIST SPECIFICATION FOR OBJECT COLLECTION (CLASSIFIED BY THE */
/* USERS AS CATEGORY 1) NAME 2ND LEVEL QUALIFIERS.                    */
/*****/

FILTLIST CAT1DATA  INCLUDE(*.CAT1*.**)

/*****/
/* FILTER LIST SPECIFICATION FOR OBJECT COLLECTION (CLASSIFIED BY THE */
/* USERS AS CATEGORY 2) NAME 2ND LEVEL QUALIFIERS.                    */
/*****/

FILTLIST CAT2DATA  INCLUDE(*.CAT2*.**)

/*****/
/* FILTER LIST SPECIFICATION FOR ACS ENVIRONMENTS FOR OBJECT DATA.  */
/*****/

FILTLIST OBJENV    INCLUDE('STORE','CHANGE','CTRANS')

```

Figure 41 (Part 1 of 2). Sample Storage Group ACS Routine for Data Sets and Objects

```

/*****/
/* START OF SELECT-WHEN-OTHERWISE GROUP OF STATEMENTS TO ASSIGN VALID */
/* STORAGE GROUPS TO OBJECTS OR OBJECT COLLECTIONS. */
/*****/

SELECT

/*****/
/* START OF SELECT-WHEN-OTHERWISE GROUP OF STATEMENTS TO ASSIGN A */
/* STORAGE GROUP TO AN OBJECT OR OBJECT COLLECTION, WHEN THE ACS */
/* ENVIRONMENT IS "STORE", "CHANGE", OR "CTRANS". */
/* */
/* NOTE: CODING OF STATEMENT HAS CHANGED TO USE FILTER LIST NAME */
/* "OBJENV". GTB - 90/12/03 */
/*****/

WHEN (&ACSENVIR = &OBJENV)

SELECT

/*****/
/* ASSIGN "OBJ00" STORAGE GROUP TO AN OBJECT, WHEN THE OBJECT BELONGS */
/* TO THE USER CATEGORY 1 OBJECT COLLECTION. */
/*****/

WHEN (&DSN = &CAT1DATA)
SET &STORGRP = 'OBJ00'

/*****/
/* ASSIGN "OBJ01" STORAGE GROUP TO AN OBJECT, WHEN THE OBJECT BELONGS */
/* TO THE USER CATEGORY 2 OBJECT COLLECTION. */
/*****/

WHEN (&DSN = &CAT2DATA)
SET &STORGRP = 'OBJ01'

/*****/
/* ASSIGN "OBJ02" STORAGE GROUP TO AN OBJECT, WHEN THE OBJECT BELONGS */
/* TO AN OBJECT COLLECTION OTHER THAN CATEGORY 1 OR 2. */
/* */
/* NOTE: CODING OF STATEMENT HAS CHANGED TO INCLUDE REPLACE A WHEN */
/* STATEMENT WITH AN OTHERWISE STATEMENT. GTB - 90/12/03 */
/*****/

OTHERWISE SET &STORGRP = 'OBJ02'

/*****/
/* END OF SELECT-WHEN-OTHERWISE GROUP OF STATEMENTS TO ASSIGN A VALID */
/* STORAGE GROUP TO AN OBJECT OR OBJECT COLLECTION. */
/*****/

END

END /* END OF SELECT-WHEN-OTHERWISE GROUP OF STATEMENTS */

```

Figure 41 (Part 2 of 2). Sample Storage Group ACS Routine for Data Sets and Objects

_____ End of General-use programming interface _____

Backup and Recovery Strategies

Using OAM for Backup and Recovery

You can use management class to provide OAM automatic backup of individual objects in an object storage group. OAM uses the AUTOBACKUP attribute in an object's management class definition. OAM does not provide a full volume dump function, but does provide automatic recovery of objects from object backups in the event of an optical volume failure.

You can use the optical disk recovery utility (an OAM function) to recover optical disks or volumes. The utility performs the necessary steps to recover the contents of an optical disk to a replacement optical disk.

The recovery of a group optical disk requires access to the backup volumes that contain the backup copies of the objects to be recovered, including tape volumes. The recovery of a backup optical disk requires access to the group optical or tape volumes that contain the objects to be recovered.

Use either of the following methods to recover an optical volume and disk:

- Issue the MODIFY OAM,START,RECOVERY operator command to initiate recovery. You must supply one of the two volume serial numbers associated with the disk to be recovered.
- Enter the RECOVER line operator against the volume listed on the ISMF Volume List panel.

Using DB2 Utilities for Backup and Recovery

Objects and object directories are stored in DB2 table spaces on DASD. DB2 utilities produce image copies of the objects and object directories. DB2 also uses activity logs and boot strap data sets as part of its recovery process. This section covers, in general terms, how database volumes can be handled by DFSMSHsm and DFSMSdss in conjunction with database utilities. DB2 databases used for storing objects and object directories would fall under the database storage group category.

IMAGECOPY

Take a regular backup copy of each of the table spaces using the DB2 IMAGECOPY utility. Determine the period between image copies, based on the use of the table spaces in an object storage group. Also, periodically run the DB2 MERGECOPY utility to consolidate incremental image copies with the latest complete image copy taken, and to create a new *base-level* image copy of the table space. This makes recovery easier than having to run either multiple image copies, or several MERGECOPY jobs to obtain the latest backup version.

RECOVER

You use the DB2 RECOVER utility to recover data. It gets data from the backup copies, and then adds the most recent changes from the recovery log. Each change made to the data in the table space is recorded in the recovery log. Therefore, recovery of a table space is done by merging the recovery log with the most recent full image copy of the table space to be recovered. This results in the table space being recovered to its last point of change, and therefore, ensures that data integrity is achieved.

If a DASD volume in an object storage group fails, you should identify all the table spaces that existed on the volume, and then recover each one individually using the DB2 utilities for recovery.

For more information about recovering DB2 table spaces, see the *DB2 Administration Guide*.

Abbreviations

The following acronyms and abbreviations are defined as they are used in the MVS/ESA Storage Management Library. If you do not find the abbreviation you are looking for, see *Dictionary of Computing*, New York: McGraw-Hill, 1994.

This list can include acronyms and abbreviations from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.
- The *Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

ABARS.	Aggregate backup and recovery support.
ACDS.	Active control data set.
ACS.	Automatic class selection.
BCDS.	Backup control data set.
BCS.	Basic catalog structure.
CHPID.	Channel path identifier.
CLIST.	Command list.
COMMDS.	Communications data set.
CTC.	Channel to channel.
CVOL.	Control volume.
DADSM.	Direct access device space management.
DASD.	Direct access storage device.
DCB.	Data control block.
DEDB.	Data entry database.
DFDSS.	Data Facility Data Set Services.
DFHSM.	Data Facility Hierarchical Storage Manager.
DFSORT.	Data Facility Sort.
DL/1.	Data Language/1
EREP.	Environmental Error Record Editing and Printing program.
GAC.	Global access checking.
GDG.	Generation data group.
GB.	Gigabyte; 10 ⁹ bytes.
GRS.	Global resource serialization.
HCD.	Hardware Configuration Definition.

HLQ.	High level qualifier.
IDRC.	Improved data recording capability.
I/O.	Input/output.
IMS.	Information Management System.
IPL.	Initial program load.
ISMF.	Interactive Storage Management Facility.
ISPF.	Interactive System Productivity Facility.
JCL.	Job control language.
JES.	Job entry subsystem.
KB.	Kilobyte; 1024 bytes.
LAN.	Local area network.
LLQ.	Lowest level qualifier.
MB.	Megabyte; 10 ⁶ bytes.
MCDS.	Migration control data set.
ML1.	Migration level 1 volume.
ML2.	Migration level 2 volume.
MSR.	Millisecond response.
MVS.	Multiple Virtual Storage.
MVSCP.	MVS configuration program.
NVR.	Non-VSAM volume record.
NVS.	Nonvolatile storage.
OAM.	Object access method.
OS.	Operating system.
OSAM.	Overflow sequential access method.
OSMC.	OAM storage management component.
PDSE.	Partitioned data set extended.
PTF.	Program temporary fix.
RMF.	Resource Measurement Facility.
SAF.	System Authorization Facility.
SCDS.	Source control data set.
SDSP.	Small data set packing.
SLR.	Service Level Reporter.
SMF.	System Management Facilities.
SML.	Storage Management Library.
SMP/E.	System Modification Program Extended.
SMS.	Storage Management Subsystem.
SSL.	Storage Subsystem Library.
TB.	Terabyte (10 ¹² bytes).

TCDB.	Tape configuration database.	VTOC.	Volume table of contents.
TSO/E.	Time Sharing Option/Extended.	VVDS.	VSAM volume data set.
VIO.	Virtual input/output.	VVR.	VSAM volume record.
VSAM.	Virtual storage access method.	WORM.	Write-once-read-many.

Glossary

The following terms are defined as they are used in the MVS/ESA Storage Management Library. If you do not find the term you are looking for, see *Dictionary of Computing*, New York: McGraw-Hill, 1994.

This glossary may include terms and definitions from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.
- The *Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

A

active control data set (ACDS). A VSAM linear data set that contains an SCDS that has been activated to control the storage management policy for the installation. When activating an SCDS, you determine which ACDS will hold the active configuration (if you have defined more than one ACDS). The ACDS is shared by each system that is using the same SMS configuration to manage storage. See also *source control data set* and *communications data set*.

active data. (1) Data that can be accessed without any special action by the user, such as data on primary storage or migrated data. Active data also can be stored on tape volumes. (2) For tape mount management, application data that is frequently referenced, small in size, and managed better on DASD than on tape. Contrast with *inactive data*.

aggregate backup. The process of copying an aggregate group and recovery instructions so that a collection of data sets can be recovered later as a group.

aggregate group. A collection of related data sets and control information that have been pooled to meet a defined backup or recovery strategy.

automated tape library. A device consisting of robotic components, cartridge storage areas, tape subsystems, and controlling hardware and software, together with the set of tape volumes that reside in the library and can be mounted on the library tape drives. See also *tape library*. Contrast with *manual tape library*.

automatic backup. (1) In DFSMSHsm, the process of automatically copying data sets from primary storage volumes or migration volumes to backup volumes. (2) In OAM, the process of automatically copying objects from DASD, optical, or tape volumes contained in an object storage group, to backup volumes contained in an object backup storage group.

automatic class selection (ACS) routine. A sequence of instructions for having the system assign data class, storage class, management class, and storage group for data sets, and storage class, management class, and storage groups for objects.

automatic data set protection. In MVS, a user attribute that causes all permanent data sets created by the user to be automatically defined to RACF with a discrete RACF profile.

automatic dump. In DFSMSHsm, the process of using DFSMSdss automatically to do a full-volume dump of all allocated space on a primary storage volume to designated tape dump volumes.

automatic primary space management. In DFSMSHsm, the process of deleting expired data sets, deleting temporary data sets, releasing unused space, and migrating data sets from primary storage volumes automatically.

automatic secondary space management. In DFSMSHsm, the process of automatically deleting expired migrated data sets, deleting expired records from the migration control data sets, and migrating eligible data sets from migration level 1 volumes to migration level 2 volumes.

automatic volume space management. In DFSMSHsm, the process that includes automatic primary space management and interval migration.

availability. For a storage subsystem, the degree to which a data set or object can be accessed when requested by a user.

B

backup. The process of creating a copy of a data set or object to be used in case of accidental loss.

base configuration. The part of an SMS configuration that contains general storage management attributes, such as the default management class, default unit, and default device geometry. It also identifies the systems or system groups that an SMS configuration is used to manage.

basic catalog structure (BCS). The name of the catalog structure in the integrated catalog facility environment. See also *integrated catalog facility catalog*.

C

cache fast write. A storage control capability in which the data is written directly to cache without using non-volatile storage. Cache fast write is useful for temporary data or data that is readily recreated, such as the sort work files created by DFSORT. Contrast with *DASD fast write*.

capacity planning. The process of forecasting and calculating the appropriate amount of physical computing resources required to accommodate an expected workload.

Cartridge System Tape. The base tape cartridge media used with 3480 or 3490 Magnetic Tape Subsystems. Contrast with *Enhanced Capacity Cartridge System Tape*.

class transition. An event that brings about change to an object's service-level criteria that causes ACS routines to be invoked by OAM to assign a new storage class or management class to the object.

compress. (1) To reduce the amount of storage required for a given data set by having the system replace identical words or phrases with a shorter token associated with the word or phrase. (2) To reclaim the unused and unavailable space in a partitioned data set that results from deleting or modifying members by moving all unused space to the end of the data set.

communications data set (COMMDS). The primary means of communication among systems governed by a single SMS configuration. The COMMDS is a VSAM linear data set that contains the name of the ACDS and current utilization statistics for each system-managed volume, which helps balance space among systems running SMS. See also *active control data set* and *source control data set*.

concurrent copy. A function to increase the accessibility of data by enabling you to make a consistent backup or copy of data concurrent with normal application program processing.

connectivity. (1) The considerations regarding how storage controls are joined to DASD and processors to achieve adequate data paths (and alternative data paths) to meet data availability needs. (2) In a DFSMS environment, the system status of volumes and storage groups.

convert in place. See *in-place conversion*.

D

DASD fast write. A storage control capability in which data is written concurrently to cache and nonvolatile storage and automatically scheduled for destaging to DASD. Both copies are retained in the storage control until the data is completely written to the DASD, providing data integrity equivalent to writing directly to the DASD. Use of DASD fast write for system-managed data sets is controlled by storage class attributes to improve performance. See also *dynamic cache management*. Contrast with *cache fast write*.

DASD volume. A DASD space identified by a common label and accessed by a set of related addresses. See also *volume*, *primary storage*, *migration level 1*, *migration level 2*.

data class. A collection of allocation and space attributes, defined by the storage administrator, that are used to create a data set.

Data Facility Sort (DFSORT). An IBM licensed program that is a high-speed data processing utility. DFSORT provides an efficient and flexible way to handle sorting, merging, and copying operations, as well as providing versatile data manipulation at the record, field, and bit level.

default device geometry. Part of the SMS base configuration, it identifies the number of bytes per track and the number of tracks per cylinder for converting space requests made in tracks or cylinders into bytes, when no unit name has been specified.

default management class. Part of the SMS base configuration, it identifies the management class that should be used for system-managed data sets that do not have a management class assigned.

default unit. Part of the SMS configuration base, it identifies an esoteric (such as SYSDA) or generic (such as 3390) device name. If a user fails to specify the UNIT parameter on JCL, SMS applies the default unit to all data sets having a disposition of either MOD or NEW that are *not* system-managed.

Device Support Facilities (ICKDSF). A program used for initialization of DASD volumes and track recovery.

DFSMS environment. An environment that helps automate and centralize the management of storage. This is achieved through a combination of hardware, software, and policies. In the DFSMS environment for MVS, this function is provided by MVS/ESA SP and DFSMS/MVS, DFSORT, and RACF. See also *system-managed storage*.

DFSMS/MVS. An IBM licensed program that together with MVS/ESA SP compose the base MVS/ESA oper-

ating environment. DFSMS/MVS consists of DFSMSdfp, DFSMSdss, DFSMSHsm, and DFSMSrmm.

DFSMSdfp. A DFSMS/MVS functional component that provides functions for storage management, data management, program management, device management, and distributed data access.

DFSMSdss. A DFSMS/MVS functional component used to copy, move, dump, and restore data sets and volumes.

DFSMSHsm. A DFSMS/MVS functional component used for backing up and recovering data, and managing space on volumes in the storage hierarchy.

DFSMSHsm-managed volume. (1) A primary storage volume, which is defined to DFSMSHsm but which does not belong to a storage group. (2) A volume in a storage group, which is using DFSMSHsm automatic dump, migration, or backup services. Contrast with *system-managed volume* and *DFSMSrmm-managed volume*.

DFSMSHsm-owned volume. A storage volume on which DFSMSHsm stores backup versions, dump copies, or migrated data sets.

DFSMSrmm. A DFSMS/MVS functional component that manages removable media.

DFSMSrmm-managed volume. A tape or optical volume that is defined to DFSMSrmm. Contrast with *system-managed volume* and *DFSMSHsm-managed volume*.

dictionary. A table that associates words, phrases, or data patterns to shorter tokens. The tokens are used to replace the associated words, phrases, or data patterns when a data set is compressed.

disaster recovery. A procedure for copying, storing, and recovering data essential to an installation's business in a secure location, and for recovering that data in the event of a catastrophic failure at the installation. Contrast with *vital records*.

drive definition. A set of attributes used to define an optical disk drive as a member of a real optical library or pseudo optical library.

dual copy. A high availability function made possible by nonvolatile storage in some models of the 3990 Storage Control. Dual copy maintains two functionally identical copies of designated DASD volumes in the logical 3990 subsystem, and automatically updates both copies every time a write operation is issued to the dual copy logical volume.

dummy storage group. A type of storage group that contains the serial numbers of volumes no longer connected to a system. Dummy storage groups allow existing JCL to function without having to be changed. See also *storage group*.

dump class. A set of characteristics that describes how volume dumps are managed by DFSMSHsm.

duplexing. The process of writing two sets of identical records in order to create a second copy of data.

dynamic cache management. A function that automatically determines which data sets will be cached based on the 3990 subsystem load, the characteristics of the data set, and the performance requirements defined by the storage administrator.

E

Enhanced Capacity Cartridge System Tape. Cartridge system tape with increased capacity that can only be used with 3490E tape subsystems. Contrast with *Cartridge System Tape*.

esoteric unit name. A name used to define a group of devices having similar hardware characteristics, such as TAPE or SYSDA. Contrast with *generic unit name*.

expiration. The process by which data sets or objects are identified for deletion because their expiration date or retention period has passed. On DASD, data sets and objects are deleted. On tape, when all data sets have reached their expiration date, the tape volume is available for reuse.

extended format. The format of a data set that has a data set name type (DSNTYPE) of EXTENDED, for example, extended sequential and extended key-sequenced data sets. Data sets in extended format can be striped or compressed.

extended sequential data set. A sequential data set that is structured logically the same as a physical sequential data set but that is stored in a different physical format. Extended sequential data sets consist of one or more stripes. See also *stripe* and *striping*.

F

filtering. The process of selecting data sets based on specified criteria. These criteria consist of fully- or partially-qualified data set names or of certain data set characteristics.

G

generic unit name. A name assigned to a class of devices with the same geometry (such as 3390). Contrast with *esoteric unit name*.

global access checking. In RACF, the ability to allow an installation to establish in-storage table of default values for authorization levels for selected resources. RACF refers to this table prior to performing normal RACHECK processing, and grants the request without performing a RACHECK if the requested access authority does not exceed the global value. Global access checking can grant the user access to the resource, but it cannot deny access.

global resource serialization (GRS). A component of MVS/ESA SP used for sharing system resources and for converting DASD reserve volumes to data set enqueues.

H

Hardware Configuration Definition (HCD). An interactive interface in MVS/ESA SP that enables an installation to define hardware configurations from a single point of control.

hierarchical file system (HFS) data set. A data set that contains a POSIX-compliant hierarchical file system, which is a collection of files and directories organized in a hierarchical structure, that can be accessed using the OpenEdition MVS facilities.

I

implementation by milestone. A conversion approach that allows for a staged conversion of your installation's data to system-managed storage on DASD, tape, or optical devices.

improved data recording capability (IDRC). A recording mode that can increase the effective cartridge data capacity and the effective data rate when enabled and invoked. IDRC is always enabled on the 3490E tape subsystem.

inactive data. (1) A copy of active data, such as vital records or a backup copy of a data set. Inactive data is never changed, but can be deleted or superseded by another copy. (2) In tape mount management, data that is written once and never used again. The majority of this data is point-in-time backups. (3) Objects infrequently accessed by users and eligible to be moved to the optical library or shelf. Contrast with *active data*.

indexed VTOC. A volume table of contents with an index that contains a list of data set names and free space information, which allows data sets to be located more efficiently.

in-place conversion. The process of bringing a volume and the data sets it contains under the control of SMS without data movement, using DFSMSdss.

integrated catalog facility catalog. A catalog that is composed of a basic catalog structure (BCS) and its related volume tables of contents (VTOCs) and VSAM volume data sets (VVDSs). See also *basic catalog structure* and *VSAM volume data set*.

Interactive Storage Management Facility (ISMF).

The interactive interface of DFSMS/MVS that allows users and storage administrators access to the storage management functions.

interval migration. In DFSMSHsm, migration that occurs automatically when the high threshold of occupancy is exceeded on a primary storage volume. Data sets are moved from the volume, largest data set first, until the low threshold of occupancy is reached.

M

management class. A collection of management attributes, defined by the storage administrator, used to control the release of allocated but unused space; to control the retention, migration, and back up of data sets; to control the retention and back up of aggregate groups, and to control the retention, back up, and class transition of objects.

manual tape library. A set of tape drives defined as a logical unit by the installation together with the set of system-managed volumes which may be mounted on those drives. See also *tape library*. Contrast with *automated tape library*.

migration. The process of moving unused data to lower cost storage in order to make space for high-availability data. The data must be recalled to be used again.

migration level 1. DFSMSHsm-owned DASD volumes that contain data sets migrated from primary storage volumes. The data can be compressed. See also *storage hierarchy*. Contrast with *primary storage* and *migration level 2*.

migration level 2. DFSMSHsm-owned tape or DASD volumes that contain data sets migrated from primary storage volumes or from migration level 1 volumes. The data can be compressed. See also *storage hierarchy*. Contrast with *primary storage* and *migration level 1*.

MVS configuration program (MVSCP). A single-step, batch program that is used to define input/output configurations to MVS.

MVS/ESA. An MVS operating system environment that supports ESA/370.

MVS/ESA SP. An IBM licensed program used to control the MVS operating system. MVS/ESA SP together with DFSMS/MVS compose the base MVS/ESA operating environment.

N

nondisruptive installation. A capability that allows users to access data in an existing storage subsystem while installing additional devices in the subsystem.

nonvolatile storage (NVS). Additional random access electronic storage with a backup battery power source, available with some storage controls, used to retain data during a power failure. Nonvolatile storage, accessible from all storage directors, stores data during DASD fast write and dual copy operations.

O

OAM-managed volumes. Optical or tape volumes controlled by the object access method (OAM).

object. A named byte stream having no specific format or record orientation.

object access method (OAM). An access method that provides storage, retrieval, and storage hierarchy management for objects and provides storage and retrieval management for tape volumes contained in system-managed libraries.

object backup storage group. A type of storage group that contains optical or tape volumes used for backup copies of objects. See also *storage group*.

object directory tables. A collection of DB2 tables that contain information about the objects that have been stored in an object storage group.

object storage group. A type of storage group that contains optical or DASD volumes used for objects. See also *storage group*.

object storage hierarchy. A three-level hierarchy consisting of objects stored in DB2 table spaces on DASD, on optical or tape volumes that reside in a library, and on optical or tape volumes that reside on a shelf. See also *storage hierarchy*.

object storage tables. A collection of DB2 tables that contain objects.

OpenEdition MVS. A feature of MVS/ESA SP that allows applications that conform to POSIX standards to operate or access data on an MVS/ESA system.

optical disk drive. The mechanism used to seek, read, and write data on an optical disk. An optical disk drive can be operator-accessible, such as the 3995 Optical Library Dataserver, or stand-alone, such as the 9346 or 9347 optical disk drives.

optical library. A storage device that houses optical drives and optical cartridges, and contains a mechanism for moving optical disks between a cartridge storage area and optical disk drives.

optical volume. Storage space on an optical disk, identified by a volume label. See also *volume*.

P

partitioned data set (PDS). A data set on direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data.

partitioned data set extended (PDSE). A system-managed data set that contains an indexed directory and members that are similar to the directory and members of partitioned data sets. A PDSE can be used instead of a partitioned data set.

performance. (1) A measurement of the amount of work a product can produce with a given amount of resources. (2) In a DFSMS environment, a measurement of effective data processing speed with respect to objectives set by the storage administrator. Performance is largely determined by throughput, response time, and system availability.

permanent data set. A user-named data set that is normally retained for longer than the duration of a job or interactive session. Contrast with *temporary data set*.

pool storage group. A type of storage group that contains system-managed DASD volumes. Pool storage groups allow groups of volumes to be managed as a single entity. See also *storage group*.

primary space allocation. Amount of space requested by a user for a data set when it is created. Contrast with *secondary space allocation*.

primary storage. A DASD volume available to users for data allocation. The volumes in primary storage are called primary volumes. See also *storage hierarchy*. Contrast with *migration level 1* and *migration level 2*.

pseudo optical library. A set of shelf-resident optical volumes associated with either a stand-alone or an

operator-accessible optical disk drive; see also *real optical library*.

R

real optical library. Physical storage device that houses optical disk drives and optical cartridges, and contains a mechanism for moving optical disks between a cartridge storage area and optical disk drives. Contrast with *pseudo optical library*.

recovery. The process of rebuilding data after it has been damaged or destroyed, often by using a backup copy of the data or by reapplying transactions recorded in a log.

removable media library. The volumes that are available for immediate use, and the shelves where they could reside.

Resource Access Control Facility (RACF). An IBM-licensed program that provides for access control by identifying and verifying the users to the system, authorizing access to protected resources, logging the detected unauthorized attempts to enter the system, and logging the detected accesses to protected resources.

Resource Measurement Facility (RMF). An IBM licensed program that measures selected areas of system activity and presents the data collected in the format of printed reports, system management facilities (SMF) records, or display reports. Use RMF to evaluate system performance and identify reasons for performance problems.

S

secondary space allocation. Amount of additional space requested by the user for a data set when primary space is full. Contrast with *primary space allocation*.

service-level agreement. (1) An agreement between the storage administration group and a user group defining what service-levels the former will provide to ensure that users receive the space, availability, performance, and security they need. (2) An agreement between the storage administration group and operations defining what service-level operations will provide to ensure that storage management jobs required by the storage administration group are completed.

Service Level Reporter (SLR). An IBM licensed program that provides the user with a coordinated set of tools and techniques and consistent information to help manage the data processing installation. For example, SLR extracts information from SMF, IMS, and CICS logs, formats selected information into tabular or graphic

reports, and gives assistance in maintaining database tables.

shelf. A place for storing removable media, such as tape and optical volumes, when they are not being written to or read.

shelf location. A single space on a shelf where you store removable media.

small data set packing (SDSP). In DFSMSHsm, the process used to migrate data sets that contain equal to or less than a specified amount of actual data. The data sets are written as one or more records into a VSAM data set on a migration level 1 volume.

small-data-set-packing data set. In DFSMSHsm, a VSAM key-sequenced data set allocated on a migration level 1 volume and containing small data sets that have been migrated.

SMS complex. A collection of systems or system groups that share a common configuration. All systems in an SMS complex share a common active control data set (ACDS) and a communications data set (COMMDS). The systems or system groups that share the configuration are defined in the SMS base configuration.

SMS configuration. A configuration base, Storage Management Subsystem class, group, library, and drive definitions, and ACS routines that the Storage Management Subsystem uses to manage storage. See also *base configuration* and *source control data set*.

source control data set (SCDS). A VSAM linear data set containing an SMS configuration. The SMS configuration in an SCDS can be changed and validated using ISMF. See also *active control data set* and *communications data set*.

storage administration group. A centralized group within the data processing center that is responsible for managing the storage resources within an installation.

storage administrator. A person in the data processing center who is responsible for defining, implementing, and maintaining storage management policies.

storage class. A named list of storage attributes that identify performance goals and availability requirements, defined by the storage administrator, used to select a device that can meet those goals and requirements.

storage control. The component in a storage subsystem that handles interaction between processor channel and storage devices, runs channel commands, and controls storage devices.

storage director. In a 3990 storage control, a logical entity consisting of one or more physical storage paths

in the same storage cluster. In a 3880, a storage director is equivalent to a storage path.

storage group. A collection of storage volumes and attributes, defined by the storage administrator. The collections can be a group of DASD volumes or tape volumes, or a group of DASD, optical, or tape volumes treated as a single object storage hierarchy. See also *VIO storage group*, *pool storage group*, *tape storage group*, *object storage group*, *object backup storage group*, and *dummy storage group*.

storage group category. A grouping of specific storage groups which contain the same type of data. This concept is analogous to storage pools in a non-system-managed environment.

storage hierarchy. An arrangement of storage devices with different speeds and capacities. The levels of the storage hierarchy include main storage (memory, DASD cache), primary storage (DASD containing uncompressed data), migration level 1 (DASD containing data in a space-saving format), and migration level 2 (tape cartridges containing data in a space-saving format). See also *primary storage*, *migration level 1*, *migration level 2*, and *object storage hierarchy*.

storage location. A location physically separate from the removable media library where volumes are stored for disaster recovery, backup, and vital records management.

storage management. The task of using storage, such as DASD, tape, or optical devices, to retain and deliver data to applications.

Storage Management Subsystem (SMS). A DFSMS/MVS facility used to automate and centralize the management of storage. Using SMS, a storage administrator describes data allocation characteristics, performance and availability goals, backup and retention requirements, and storage requirements to the system through data class, storage class, management class, storage group, and ACS routine definitions.

storage subsystem. A storage control and its attached storage devices. See also *tape subsystem*.

stripe. The portion of a striped data set (for example, an extended sequential data set) that resides on one disk. The records in that portion are not necessarily logically consecutive. The system distributes records among the stripes such that the disks can be read or written simultaneously to gain better performance. See also *extended sequential data set*.

striping. A software implementation of a disk array that distributes data sets across multiple volumes to improve performance.

sysplex. A systems complex that consists of multiple systems coupled together by hardware elements and system services. The systems must be defined by the COUPLExx members of SYS1.PARMLIB as being a sysplex.

system data. The data sets required by MVS or its subsystems for initialization and control.

system group. All systems that are part of the same sysplex and are running the Storage Management Subsystem with the same configuration, minus any systems in the sysplex that are explicitly defined in the SMS configuration.

system-managed data set. A data set that has been assigned a storage class.

system-managed storage. Storage managed by the Storage Management Subsystem. SMS attempts to deliver required services for availability, performance, space, and security to applications. See also *DFSMS environment*.

system-managed tape library. A collection of tape volumes and tape devices, defined in the tape configuration database. A system-managed tape library can be automated or manual. See also *tape library*.

system-managed volume. A DASD, optical, or tape volume that belongs to a storage group. Contrast with *DFSMSshsm-managed volume* and *DFSMSrmm-managed volume*.

system management facilities (SMF). A component of MVS/ESA SP that collects input/output (I/O) statistics, provided at the data set and storage class levels, which helps you monitor the performance of the direct access storage subsystem.

system programmer. A programmer who plans, generates, maintains, extends, and controls the use of an operating system and applications with the aim of improving overall productivity of an installation.

T

tape configuration database. One or more volume catalogs used to maintain records of system-managed tape libraries and tape volumes.

tape library. A set of equipment and facilities that support the installation's tape environment. This may include tape storage racks, a set of tape drives, and a set of related tape volumes which can be mounted on those drives. See also *system-managed tape library*.

tape librarian. The person who manages the tape library.

Tape Library Dataserver. A hardware device that maintains the tape inventory associated with a set of tape drives. An automated tape library dataserver also manages the mounting, dismounting, and storage of tapes.

tape mount management. The methodology used to optimize tape subsystem operation and use, consisting of hardware and software facilities used to manage tape data efficiently.

tape storage group. A type of storage group that contains system-managed private tape volumes. The tape storage group definition specifies the system-managed tape libraries that can contain tape volumes. See also *storage group*.

tape subsystem. A magnetic tape subsystem consisting of controller and devices, which allows users to store data on tape cartridges. Examples of tape subsystems include the IBM 3490 and 3490E Magnetic Tape Subsystems.

tape volume. Storage space on tape, identified by a volume label, which contains data sets or objects and available free space. A tape volume is the recording space on a single tape cartridge or reel. See also *volume*.

temporary data set. An uncataloged data set whose name begins with & or &&, that is normally used only for the duration of a job or interactive session. Contrast with *permanent data set*.

threshold. A storage group attribute that controls the space usage on DASD volumes. The *low threshold* is used during primary space management to determine the amount of free space to provide on the volume. The *high threshold* is used to determine candidate volumes for new data set allocations, and to determine if data sets must be migrated off of a volume during interval migration.

U

unmovable data set. A DASD data set required by an application, which cannot be moved. Unmovable data sets need to be identified to the system so that they are not relocated. They can be identified by allocating the data set in absolute tracks or by allocating a data set with the data set organization that includes the unmovable attribute. For example, data sets allocated as PSU, POU, DAU, or ABSTR are considered unmovable.

use attribute. (1) The attribute assigned to a DASD volume that controls when the volume can be used to allocate new data sets; use attributes are *public*,

private, and *storage*. (2) For system-managed tape volumes, use attributes are *scratch* and *private*.

user group. A group of users in an installation who represent a single department or function within the organization.

user group representative. A person within a user group who is responsible for representing the user group's interests in negotiations with the storage administration group.

V

validate. To check the completeness and consistency of an individual ACS routine or an entire SMS configuration.

virtual input/output (VIO) storage group. A type of storage group that allocates data sets to paging storage, which simulates a DASD volume. VIO storage groups do not contain any actual DASD volumes. See also *storage group*.

vital records. A data set or volume maintained for the purpose of meeting an externally imposed retention requirement, such as a legal requirement. Contrast with *disaster recovery*.

vital record specification. Policies defined to manage the retention and movement of data sets and volumes for disaster recovery and vital records purposes.

volume. The storage space on DASD, tape, or optical devices, which is identified by a volume label. See also *DASD volume*, *optical volume*, and *tape volume*.

volume status. In the Storage Management Subsystem, indicates whether the volume is fully available for system management:

- "Initial" indicates that the volume is not ready for system management because it contains data sets that are ineligible for system management.
- "Converted" indicates that all of the data sets on a volume have an associated storage class and are cataloged in an integrated catalog facility catalog.
- "Non-system-managed" indicates that the volume does not contain any system-managed data sets and has not been initialized as system-managed.

VSAM volume data set (VVDS). A data set that describes the characteristics of VSAM and system-managed data sets residing on a given DASD volume; part of an integrated catalog facility catalog. See also *basic catalog structure* and *integrated catalog facility catalog*.

Index

Numerics

- 3390 Model 9 19
- 3990 Model 3 Storage Control 7
 - cache statistics 39
 - concurrent copy capability 42
 - dual copy 41
 - dynamic cache management 39
 - sequential data striping 40
 - SMS 38
- 3990 Model 6 Storage Control 7, 19
- 3995 Optical Library Dataserver 19
- 9121 processor 20

A

- abbreviations 111
- access method services
 - adding volumes to tape libraries 94
 - defining tape libraries 89
 - defining volume catalogs 88
- ACDS (active control data set)
 - description 13
- acronyms 111
- ACS (automatic class selection) routine
 - description 11
 - storage class
 - example for temporary data sets 59—61
 - storage group
 - example for DASD data 58—59
 - example for object data 106—108
 - example for tape data 96—97
 - example for tape mount management 85—86
- adding storage groups 63
- adding volumes to storage groups 63
- aggregate group
 - compatibility between DFHSM and DFSMSHsm 19
 - overview 11
- allocation
 - data class, using 6
 - DFSORT temporary data sets 26
- ARCMVEXT installation exit 79
- attribute
 - compaction 95
 - object storage group 101
 - pool storage group 48
 - tape storage group 92
- automatic backup 49
- automatic dumping 49
- automatic incremental backup 70, 71
- automatic migration 49

- availability
 - configuring DASD for 41
 - copying concurrently 42
 - data set 69
 - dual copy 41
 - hardware 69
 - monitoring for storage groups 69

B

- backup procedures
 - incremental 73
 - strategies for different data types 73
- backup volumes
 - volume serial numbers 33
 - VTOC size 45
- base configuration 13
- batch data set 24
- buffer, tape mount management 83, 84

C

- cache
 - dynamic cache management 39
 - SMS 38
 - statistics 39
- CADAM (Computer-Aided Design and Manufacturing System) 26
- catalog
 - sharing
 - between MVS/ESA and MVS/XA 14
 - in JES3 installation 14
 - volume 88
- category, storage group 23
- CICS (Customer Information Control System) 26
- class transition 104
- coexistence
 - between MVS/ESA and MVS/XA
 - catalog sharing 14
 - DASD sharing 16
 - data set
 - among multisite systems 14
 - multisystem environment 17
- COMMDS (communications data set)
 - description 13
- COMPACTION data class attribute 95
- compressing partitioned data sets 78
- condensing partitioned data sets 78
- configuration planning
 - concurrent copy 42
 - dual copy 41
 - dynamic I/O configuration 43

- configuration planning (*continued*)
 - for availability 41
- console name, tape library 91
- control data set
 - Storage Management Subsystem
 - ACDS (active control data set) 13
 - COMMDS (communications data set) 13
 - overview 13
 - SCDS (source control data set) 13
- converted volume status 46
- converting to the DFSMS environment
 - with volume pools 29
 - without volume pools 27
- CONVERTV command 46
- cross-system sharing 35

D

- DASD (direct access storage device)
 - availability 41, 42
 - capacity 32
 - combining with optical device in storage group 33
 - dynamic I/O configuration 43
 - geometry 32
 - naming conventions 84
 - number in storage group 36
 - sharing 35
 - sharing DASD 16
 - volume
 - selection 9
 - status 46
- data class
 - compaction 95
 - overview 6
 - striped sequential data set 40
 - tape recording method 95
- data set
 - ACDS (active control data set) 13
 - allocation
 - data class 6
 - availability 69
 - backup 73
 - batch 24
 - cache statistics 39
 - coexistence
 - among multisite systems 14
 - multisystem environment 17
 - shared DASD 16
 - COMMDS (communications data set) 13
 - compressing 78
 - database 26
 - incremental backup 73
 - interactive 24
 - ISPF/PDF libraries 24
 - large 25
 - listing by group 32

- data set (*continued*)
 - migrating, recalling 75
 - multivolume 24
 - releasing unused space 78
 - SCDS (source control data set) 13
 - sharing 14
 - storage class
 - example ACS routine for temporary data 59—61
 - storage group
 - example ACS routine for DASD data 58—59
 - example ACS routine for tape data 96—97
 - example ACS routine for tape mount management 85—86
 - system 24
 - temporary 26
 - transferring between systems 14
 - TSO 24
 - VSAM 24
- database administrator 26
- database storage group 26, 73
- DCOLLECT command
 - capacity planning and trend reporting 66
 - examples 64
- DEFINE USERCATALOG command
 - volume catalogs 88
- DEFRAG command 79
- deleting storage groups 63
- deleting volumes from storage groups 63
- Device Support Facilities INIT command 44
- DFDSS
 - coexistence between releases 19
- DFHSM
 - coexistence between releases 19
- DFSMS environment
 - benefits 2
 - coexistence between MVS/ESA and MVS/XA 14
 - managing data 4
 - product requirements 1
- DFSMS/MVS
 - description 1
 - DFDSS coexistence 19
 - DFHSM coexistence 19
 - sharing
 - data sets 14
- DFSMSdfp
 - description 1
- DFSMSdss
 - coexistence with DFDSS 19
 - concurrent copy 42
 - converting volumes to system-managed 46
 - CONVERTV command
 - volume status definitions 46
 - data set backup 73
 - defragmentation 79
 - description 2

DFSMSHsm
 automatic incremental backup 70, 73
 automatic storage group processing 49
 calling DFSMSdss 73
 coexistence with DFHSM 19
 description 2
 FREEVOL command 77
 incremental backup 71
 migrated data, increasing compaction 86
 migration 75
 recall 75
 recycle processing 78
 space management 75
 space management volume installation exit 79
 storage group thresholds 50—53
 volume recovery 71
DFSMSrmm
 adding volumes to a tape library 94
 data set information 96
 description 2
 reserved location names 90
 tape inventory 98
 tape libraries 98
DFSORT (Data Facility Sort)
 allocation considerations 26, 59
 description 2
 temporary data sets 59
disaster recovery
 aggregate group
 overview 11
Distributed File Manager/MVS
 data sets created by 12
 drive, optical 103
DSCB (data set control block)
 compatibility between MVS/XA and MVS/ESA 16
 estimating VTOC size 45
 dump class 49
 dynamic cache management 39

E
ERBRMF member of SYS1.PARMLIB 69
extended sequential data sets
 storage group 32

F
 fragmentation index 79
 fragmentation, reducing 79

G
 glossary 113
GRS (global resource serialization) 35
 guaranteed backup frequency 49

H
 hardware availability 69
Hardware Configuration Definition 43, 48
 hardware considerations 19
 high threshold 50

I
ICKDSF, initializing volumes 44
IEASYS00 member of SYS1.PARMLIB 47
 improved data recording capability 86
indexed VTOC (volume table of contents)
 allocating 43
 placement 45
 size 45
INIT command 44
 initial volume status 46
 initializing volumes 43
 interactive data set 24
 interval migration 82
ISMF (Interactive Storage Management Facility)
 compressing partitioned data sets 78
 condensing partitioned data sets 78
 initializing volumes 44
 listing data sets 32
 managing space 78
 RELEASE function 78
 releasing unused space 78
 tape libraries, defining 89

L
 label, tape 94
large data set
 definition 25
 space restrictions 25
 storage group 25
large storage group
 backup and recovery 73
library, optical disk
 determining requirements 102
library, tape
 adding volumes 94
 advantages 87
 console name 91
 defining 89
 library ID 91
 naming conventions 90
 types 87
 volume attributes 91
 low threshold 51

M
 maintaining storage groups 63

- management class
 - objects 104
 - overview 8
- migration level 1 volumes
 - volume serial numbers 33
- multisite systems 14
- multisystem environment
 - catalog sharing 14
 - DASD sharing
 - between MVS/ESA and MVS/XA 16
 - description 17
 - system group 18
- multivolume data set 24
- MVS/ESA
 - coexistence with MVS/XA 14
 - products for the DFSMS environment 1
 - sharing
 - data sets 14
 - sharing DASD 17
 - VTOC compatibility with MVS/XA 16
- MVS/XA
 - coexistence with MVS/ESA 14
 - sharing
 - data sets 14
 - VTOC compatibility with MVS/ESA 16

N

- naming conventions
 - DASD volumes 33, 84
 - storage group 34, 84
 - tape 94
 - tape library 90
 - tape volumes 95
- NetView
 - transferring data between systems 14
- non-system-managed DASD volumes 46

O

- OAM (object access method)
 - library control system 87
 - management class
 - class transition attributes 104
 - managing space 99
- object
 - backup and recovery 109
 - data class 7
 - determining management class requirements 104
 - determining storage class requirements 103
 - determining storage group requirements 100
 - example storage group ACS routine 106—108
 - how stored 99
 - optical volumes 104
 - storage class 5, 8
 - storage groups 99

- object (*continued*)
 - tape volumes 103
- optical disk drive
 - determining requirements 103
 - storage group 101
- optical library
 - determining requirements 102
 - storage group 101
- optical volumes 104
- overflow storage 54

P

- partitioned data set
 - compressing 78
 - condensing 78
- partitioned data sets
 - releasing unused space with ISMF 78
- performance
 - monitoring for storage groups 68
- primary storage group
 - backup and recovery 73

R

- RACF (Resource Access Control Facility)
 - description 2
 - releasing unused space 78
- RMF (Resource Measurement Facility)
 - DFSMSHsm delay report 68
 - monitoring performance for storage pools 68

S

- SCDS (source control data set)
 - description 13
- serialization between systems 17
- service-level agreement 64
- sharing
 - DASD volumes 16
 - data sets 14, 35
 - storage groups 35
 - volumes 35
- SLR (Service Level Reporter)
 - monitoring performance for storage pools 68
- small data set packing
 - VTOC size 45
- SMF (System Management Facilities)
 - Type 42 records 39
- SMFPRM member of SYS1.PARMLIB 69
- SMS (Storage Management Subsystem)
 - 3990 storage control 38
 - aggregate group
 - overview 11
 - base configuration 13
 - cache 38

SMS (Storage Management Subsystem) (*continued*)

- cache statistics 39
- control data sets
 - ACDS (active control data set) 13
 - COMMDS (communications data set) 13
 - overview 13
 - SCDS (source control data set) 13
- data class
 - overview 6
- DFSORT temporary data sets 26
- dynamic cache management 39
- management class
 - overview 8
- managing data 4
- storage class
 - See also* storage class
 - overview 7
- storage group
 - See also* storage group
 - overview 9
- space management volume installation exit 79
- spill buffer, tape mount management 83, 84
- spill storage 54
- spill volumes, clearing 75
- storage class
 - ACS routine
 - example for temporary data sets 59—61
 - availability goals 37
 - cache statistics 39
 - determining requirements 37
 - objects 8, 103
 - overview 7
 - performance goals 37
 - striped sequential data set 40
- storage control
 - cache statistics 39
 - concurrent copy capability 42
 - dual copy 41
 - dynamic cache management 39
 - IBM 3990 advantages 38
 - sequential data striping 40
- storage group
 - ACS routine
 - example for DASD data 58—59
 - example for tape data 96—97
 - example for tape mount management 85—86
 - adding 63
 - automatic processing 49
 - categories 23
 - combining device types 33
 - connectivity 54
 - database 26
 - defining 48
 - defining thresholds 50—53
 - deleting 63
 - device capacity 32

storage group (*continued*)

- device geometry 32
- extended sequential data sets 32
- large 25
- logical and physical requirements 31
- maintaining 63
- managing availability 69
- monitoring performance 68
- naming conventions 34, 84
- number of DASD volumes in 36
- number of volumes in 32
- object 99, 109
- overview 9
- preventing allocation failures 54
- primary 24
- sharing 35
- spill storage group 55
- tape
 - defining 92
 - determining requirements 93
 - naming conventions 94
- tape mount management
 - examples 84
- tape mount management buffers 83
- types 22
- VIO 26
- volume selection 9

storage overflow for preventing allocation failures 54

striped sequential data set

- defining 40
- storage class 40

sustained data rate 40

SYS1.PARMLIB data set

- ERBRMF member 69
- IEASYS00 member 47
- SMFPRM member 69
- volume attribute list (VATLST) 47

SYS1.VOLCAT.VGENERAL catalog 88

SYS1.VOLCAT.Vx catalog 88

system data 24

system group

- coexistence 18—19
- description 17

system-managed

- DASD volumes 46

system-managed storage

- benefits 2
- definition 1
- managing data 4

T

tape

- analyzing tape mounts 83
- capacity 95
- cataloging 96

- tape (*continued*)
 - configuration database 88
 - inventory 98
 - labels 94
 - library
 - adding volumes 94
 - advantages 87
 - automated 87, 92
 - console name 91
 - defining 89
 - DFSMSrmm 98
 - library ID 91
 - manual 87, 92
 - naming conventions 90
 - OAM library control system 87
 - types 87
 - volume attributes 91
 - naming conventions 95
 - object backup 103
 - storage group
 - defining 92
 - determining requirements 93
 - example ACS routine 96—97
 - example ACS routine for tape mount management 85—86
 - tape mount management buffers 83
- tape configuration database
 - defining 88
- tape data set
 - storage class 5
- tape mount management
 - analyzing tape mounts 83
 - buffers 83
 - DASD volume naming conventions 84
 - overview 82
 - storage group
 - example ACS routine 85—86
 - examples 84
 - naming conventions 84
- temporary data set 26
- thresholds, volume 50—53
- TSO data set 24
- TSO TRANSMIT and RECEIVE commands 14

U

- Unit Control Blocks (UCBs) 19
- unmovable data
 - backup and recovery 73

V

- VAL parameter 47
- VATLST member of SYS1.PARMLIB 47
- VIO (virtual input/output)
 - DFSORT temporary data sets 26

- VIO (virtual input/output) (*continued*)
 - storage group 26
- VOLCATALOG parameter, DEFINE USERCATALOG
 - command 88
- volume
 - adding 63
 - attribute list, updating 47
 - catalog 88
 - connectivity 54
 - converting to system-managed 46
 - DASD (direct access storage device)
 - number in storage group 36
 - selection 9
 - status during SMS conversion 46
 - thresholds 50—53
 - defragmentation 79
 - deleting 63
 - initializing 43
 - preventing allocation failures 54
 - recovery 71
 - removing 48
 - tape
 - adding to tape library 94
 - attributes 91
 - capacity 95
 - inventory 98
 - volume mount analyzer 83
- VSAM data set 24
- VTOC (volume table of contents)
 - allocating 43
 - compatibility between MVS/XA and MVS/ESA 16
 - DASD volume status indicators 46
 - placement 45
 - size 45

X

- XMIT (TSO TRANSMIT) command 14

Readers' Comments

MVS/ESA Storage Management Library Managing Storage Groups

Publication No. SC26-3125-02

Please send us your comments about this printed book, or its softcopy version on CD-ROM. You can also send us comments on product-related online help panels, if applicable.

In sending information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Note: Do not use this form to request IBM publications. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.

Mail the completed form to the address on the reverse side. For postage-paid mailing from a country other than the United States, give it to your IBM representative or to the IBM branch office serving your locality.

You can also send us comments by fax:

United States: 1-800-426-6209

You can send us comments electronically:

IBMLink from U.S.: STARPUBS at SJSVM28
IBMLink from Canada: STARPUBS at TORIBM
IBM Mail Exchange: USIB3VVD at IBMMAIL
Internet: starpubs at sjsvm28.vnet.ibm.com
or, you can use
starpubs@sjsvm28.vnet.ibm.com

Name

Address

Company or Organization

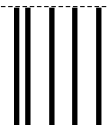
Phone No.



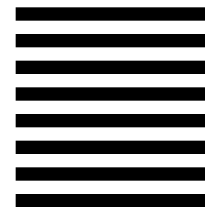
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Department M29, Building 86
International Business Machines Corporation
5600 Cottle Road
SAN JOSE CA 95123-9988



Fold and Tape

Please do not staple

Fold and Tape



Printed in U.S.A.

MVS/ESA Storage Management Library

SC26-3123 Implementing System-Managed Storage
SC26-3126 Leading a Storage Administration Group
SC26-3124 Managing Data
SC26-3125 Managing Storage Groups

SC26-3125-02



Spine information:



MVS/ESA SML

Managing Storage Groups

SC26-3125-02